IBM Tivoli OMEGAMON XE for DB2 Performance Expert on z/OS
IBM Tivoli OMEGAMON XE for DB2 Performance Monitor on z/OS
Version 5.2.0

Report Reference
IBM Tivoli OMEGAMON XE for DB2 Performance Expert on z/OS
IBM Tivoli OMEGAMON XE for DB2 Performance Monitor on z/OS
Version 5.2.0

Report Reference
Note

Before using this information and the product it supports, read the information in “Notices” on page A-1.

First edition, October 2013

This edition applies to the following releases and to all subsequent releases and modifications until otherwise indicated in new editions:

• IBM Tivoli OMEGAMON XE for DB2 Performance Expert on z/OS, version 5, release 2, modification 0 (5655-W37)
• IBM Tivoli OMEGAMON XE for DB2 Performance Monitor on z/OS, version 5, release 2, modification 0 (5655-W38)

This edition replaces SH12-6963-00.

© Copyright IBM Corporation 1985, 2013.
US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
Contents

About this publication ................................................. xix
Who should read this publication ................................. xx
Terminology used in this publication ............................... xxi
Conventions used in the OMEGAMON documentation .............. xxii
  Terminology used .................................................. xxv
  How to read syntax diagrams ..................................... xxvi
  Where to find information ....................................... xxviii
  Service updates and support information ......................... xxix
  Accessibility features ........................................... xxx
  How to send your comments .................................... xxxi

What’s new ................................................................. xxxiii

Part 1. OMEGAMON XE for DB2 PE Logs

Chapter 1. DPMLOG Execution Log .................................. 1-1
  The DPMLOG Execution Log Header .............................. 1-2
  Field Descriptions ................................................ 1-3

Chapter 2. Exception Log .............................................. 2-1
  How to generate an Exception Log ............................... 2-2
  Example of the Exception Log .................................... 2-3
    The Exception Log Header ..................................... 2-4
    Field Descriptions ............................................ 2-5

Chapter 3. Job Summary Log .......................................... 3-1
  How to Generate the Job Summary Log ......................... 3-2
  Example of the Job Summary log ................................ 3-3
    The Job Summary Log Header ................................ 3-4
    Field Descriptions ............................................ 3-5
  Job Summary VSAM Data Set ..................................... 3-6

Chapter 4. IFCID Frequency Distribution Log ..................... 4-1
  How to Generate an IFCID Frequency Distribution Log ........ 4-2
  Example of the IFCID Frequency Distribution Log ............. 4-3
    The IFCID Frequency Distribution Log Header ............. 4-4
    Field Descriptions ............................................ 4-5

Part 2. The Accounting Report Set

Chapter 5. Accounting Default Layouts ......................... 5-1
  Headers Used in Accounting .................................... 5-2
  How Averages Are Calculated .................................. 5-4
  Accounting Report - Short ...................................... 5-6
    General (Short Report) ....................................... 5-7
    Package General (Short Report) .............................. 5-12
    Distributed Activity Server (Short Report) ............... 5-15
    Distributed Activity Requester (Short Report) .......... 5-18
  Accounting Trace - Short ....................................... 5-20
    General (Short Trace) ......................................... 5-21
    Package General (Short Trace) .............................. 5-28
    Distributed Activity Server (Short Trace) ............... 5-30
    Distributed Activity Requester (Short Trace) .......... 5-33

© Copyright IBM Corp. 1985, 2013
Using Lock Suspension Data with Spreadsheets .................................. 33-9
Lockout Report .................................................................................. 33-12
Locking Detail Report ................................................................. 33-15

Chapter 34. Locking Trace .......................................................... 34-1
Deadlock Trace ............................................................................. 34-3
  Trace Data Specific to Deadlock Event ........................................ 34-4
  Deadlock Trace Example .......................................................... 34-7
Timeout Trace ............................................................................... 34-11
  Trace Data Specific to Timeout Event ........................................ 34-12
  Timeout Trace Example .......................................................... 34-15
Lockout Trace ............................................................................... 34-16
Lock Suspension Trace ............................................................... 34-17
  Lock Suspension Trace Example ............................................. 34-18
  Lock Suspension Events - Lock, Unlock, Change, and Notify Suspend 34-19
  Lock Suspension Events - Lock, Unlock, Change, and Notify Resume 34-21
  Lock Suspension Events - Query Suspend .................................... 34-24
  Lock Suspension Events - Query Resume .................................... 34-25
  Lock Suspension Events - Drain Suspend .................................... 34-26
  Lock Suspension Events - Drain Resume .................................... 34-27
  Lock Suspension Events - Latch Suspend .................................... 34-28
  Lock Suspension Events - Latch Resume .................................... 34-29
  Lock Detail Trace Data ............................................................ 34-30
  Example of a Lock Detail Trace ............................................... 34-31
  Lock Detail Trace - Lock Summary ........................................... 34-32
  Lock Detail Trace - Lock, Unlock, and Change Requests .............. 34-34
  Lock Detail - Query Requests .................................................. 34-37
  Lock Detail - Claim Acquire, Change, and Release ..................... 34-38
  Lock Detail - Drain Request, Pseudo, and Release ...................... 34-40
  Lock Detail - Lock Avoidance ................................................... 34-42
  Lock Detail - P-Lock Requests .................................................. 34-43
  Lock Detail - Notify Request .................................................... 34-46
  Lock Detail - Lock Escalation .................................................... 34-47

Chapter 35. The Locking File Data Set ......................................... 35-1

Part 7. The Record Trace Report Set

Chapter 36. Record Headers ....................................................... 36-1

Chapter 37. The Summary Record Trace .................................... 37-1
ACE Cross-Reference Table ......................................................... 37-3
Data Fields ................................................................................. 37-4
  Logical Unit of Work Identifiers ............................................. 37-5
  DDF Data ................................................................................. 37-6
  Accounting Token ................................................................. 37-7

Chapter 38. The Short and Long Record Traces ......................... 38-1
  The Short Record Trace .......................................................... 38-2
  The Long Record Trace .......................................................... 38-3

Chapter 39. Dump Record Trace .................................................. 39-1
Column Descriptions of the Dump Record Trace ....................... 39-3
ACE Cross-Reference Table ......................................................... 39-5
Field Descriptions of the Dump Record Trace ......................... 39-6
  Logical Unit of Work Identifiers ............................................. 39-7
  DDF Data ................................................................................. 39-8
  Accounting Token ................................................................. 39-9
Chapter 40. IFCID Record Blocks ........................................ 40-1
IFCID 001 - System Statistics ........................................ 40-19
IFCID 001 - Checkpoint and IFI Data .............................. 40-20
IFCID 001 - CPU Time Data ........................................... 40-23
IFCID 001 - DB2 Command Data ...................................... 40-25
IFCID 001 - DDF Data by Location .................................. 40-31
IFCID 001 - Destination Related Data .............................. 40-38
IFCID 001 - Global DDF Data ......................................... 40-40
IFCID 001 - IFCID Data ................................................ 40-44
IFCID 001 - Log Manager Data ...................................... 40-45
IFCID 001 - Subsystem Services Data ............................... 40-50
IFCID 001 - QSST Data ............................................... 40-55
IFCID 001 - QVAS Data ............................................... 40-56
IFCID 001 - QVLS Data ............................................... 40-57
IFCID 001 - z/OS Metrics ............................................. 40-61
IFCID 002 - DB2 Statistics ........................................... 40-63
IFCID 002 - Accelerator Data ....................................... 40-64
IFCID 002 - Buffer Pool Activity ................................... 40-69
IFCID 002 - Data Manager Data ..................................... 40-85
IFCID 002 - Data Sharing Locking Data ............................ 40-92
IFCID 002 - Dynamic SQL Statement .............................. 40-95
IFCID 002 - EDM Pool Data ......................................... 40-97
IFCID 002 - Group Buffer Pools Activity Data .................... 40-103
IFCID 002 - Locking Data ............................................ 40-112
IFCID 002 - Miscellaneous .......................................... 40-116
IFCID 002 - Nested SQL Activity ................................... 40-118
IFCID 002 - Query Parallelism ...................................... 40-120
IFCID 002 - RID List Processing .................................... 40-124
IFCID 002 - ROWID .................................................. 40-126
IFCID 002 - Service Controller Data ............................... 40-127
IFCID 002 - SQL Call Data ......................................... 40-135
IFCID 003 - Accounting ............................................... 40-144
IFCID 003 - Accelerator Data ....................................... 40-146
IFCID 003 - Buffer Manager Accounting Data .................... 40-149
IFCID 003 - Data Sharing Accounting Data ....................... 40-153
IFCID 003 - Data Sharing Locking Data ........................... 40-154
IFCID 003 - DDF Data by Location ................................ 40-156
IFCID 003 - Dynamic SQL Statement ............................. 40-165
IFCID 003 - Group Buffer Pools Activity Data .................... 40-167
IFCID 003 - IFI Class 5 Times and Data Capture ................. 40-169
IFCID 003 - Initial DB2 Requester and MVS Correlation Data ........................................ 40-171
IFCID 003 - Instrumentation Accounting Data ..................... 40-175
IFCID 003 - Instrumentation Accounting Data Overflow ......... 40-190
IFCID 003 - Locking Data ............................................ 40-193
IFCID 003 - Logging .................................................. 40-197
IFCID 003 - Miscellaneous ......................................... 40-198
IFCID 003 - Nested SQL Activity ................................... 40-199
IFCID 003 - Query Parallelism ...................................... 40-201
IFCID 003 - Resource Limit Facility ............................... 40-202
IFCID 003 - RID List Processing .................................... 40-204
IFCID 003 - Rollup Accounting Correlation Block (DB2 10 or later) ......................... 40-205
IFCID 003 - ROWID .................................................. 40-206
IFCID 003 - SQL Call Data ......................................... 40-207
IFCID 004 - Trace Start .............................................. 40-208
IFCID 005 - Trace Stop ................................................ 40-209
IFCID 006 - Read I/O Start .......................................... 40-210
IFCID 007 - Read I/O Stop .......................................... 40-212
IFCID 008 - Write I/O Synch ........................................ 40-213
IFCID 009 - Write I/O ............................................... 40-215
IFCID 010 - Write I/O Asynch ....................................... 40-216
IFCID 011 - Validate Exit ........................................... 40-218
IFCID 012 - Edit Exit to Encode ........................................... 40-219
IFCID 013 - Hash Scan Input Start ........................................ 40-220
IFCID 014 - Hash Scan End .................................................. 40-222
IFCID 015 - Index Scan Begin ............................................... 40-223
IFCID 016 - Insert Scan Begin ............................................. 40-225
IFCID 017 - Sequential Scan Begin ....................................... 40-227
IFCID 018 - Scan End .......................................................... 40-229
IFCID 019 - Edit Exit to Decode ........................................... 40-231
IFCID 020 - Lock Summary ................................................... 40-232
IFCID 021 - Lock Detail ........................................................ 40-234
IFCID 022 - Minibind .......................................................... 40-239
IFCID 023 - Utility Start ...................................................... 40-248
IFCID 024 - Utility Change .................................................. 40-252
IFCID 025 - Utility End ....................................................... 40-254
IFCID 026 - IBM Service Record ......................................... 40-257
IFCID 027 - Sort Workfile Records ..................................... 40-258
IFCID 028 - Sort Phase Detail ............................................. 40-260
IFCID 029 - EDM Request Start ......................................... 40-262
IFCID 030 - EDM Request End ............................................ 40-264
IFCID 031 - EDM Full .......................................................... 40-266
IFCID 032 - Log Wait Start .................................................. 40-269
IFCID 033 - IBM Service Record ....................................... 40-270
IFCID 034 - Log Read Start ................................................... 40-271
IFCID 035 - Log Read End ................................................... 40-272
IFCID 036 - Log Non I/O Start ............................................ 40-273
IFCID 037 - Log Non I/O End ................................................... 40-274
IFCID 038 - Active Write Start ........................................... 40-275
IFCID 039 - Active Write End ............................................. 40-276
IFCID 040 - Archive Write Start ....................................... 40-277
IFCID 041 - Archive Write End .......................................... 40-278
IFCID 042 - Checkpoint Start ............................................. 40-279
IFCID 043 - Checkpoint End ............................................... 40-280
IFCID 044 - Lock Suspend .................................................... 40-281
IFCID 045 - Lock Resume ..................................................... 40-285
IFCID 046 - IBM Service Record ....................................... 40-287
IFCID 047 - IBM Service Record ....................................... 40-288
IFCID 048 - IBM Service Record ....................................... 40-289
IFCID 049 - IBM Service Record ....................................... 40-290
IFCID 050 - IBM Service Record ....................................... 40-291
IFCID 051 - IBM Service Record ....................................... 40-292
IFCID 052 - IBM Service Record ....................................... 40-293
IFCID 053 - SQL Describe/Commit/Back/Rollback/Remote Statement ................................................. 40-294
IFCID 055 - Set SQLID .......................................................... 40-297
IFCID 056 - IBM Service Record ....................................... 40-298
IFCID 057 - IBM Service Record ....................................... 40-299
IFCID 058 - End SQL ........................................................... 40-300
IFCID 059 - Fetch Start ........................................................ 40-303
IFCID 060 - Select Start ..................................................... 40-305
IFCID 061 - Insert/Update/Delete Start ................................. 40-307
IFCID 062 - DDL Start .......................................................... 40-309
IFCID 063 - SQL Statement .................................................... 40-310
IFCID 064 - Prepare Start .................................................... 40-312
IFCID 065 - Open Cursor ...................................................... 40-314
IFCID 066 - Close Cursor .................................................... 40-317
IFCID 067 - Accounting .......................................................... 40-319
IFCID 068 - Rollback Start ................................................... 40-320
IFCID 069 - IBM Service Record ....................................... 40-321
IFCID 070 - Commit Phase 2 Start ...................................... 40-322
IFCID 071 - IBM Service Record ....................................... 40-323
IFCID 072 - Create Thread Start ....................................... 40-324
IFCID 073 - Create Thread End ............................................. 40-325
IFCID 074 - Terminate Thread Start ........................................ 40-326
IFCID 075 - Terminate Thread End ........................................... 40-327
IFCID 076 - End of Memory Start .............................................. 40-328
IFCID 077 - End of Memory End ................................................ 40-329
IFCID 078 - End of Task Start .................................................. 40-330
IFCID 079 - End of Task End .................................................... 40-331
IFCID 080 - IBM Service Record .............................................. 40-332
IFCID 081 - IBM Service Record .............................................. 40-333
IFCID 082 - Identify Start ....................................................... 40-334
IFCID 083 - Identify End ....................................................... 40-335
IFCID 084 - Prepare Start ....................................................... 40-336
IFCID 085 - Prepare End ....................................................... 40-337
IFCID 086 - Signon Start ....................................................... 40-338
IFCID 087 - Signon End ....................................................... 40-339
IFCID 088 - Synch Start ....................................................... 40-340
IFCID 089 - Synch End ....................................................... 40-341
IFCID 090 - DB2 Command Start .............................................. 40-342
IFCID 091 - Command End ..................................................... 40-343
IFCID 092 - AMS Command Start ............................................. 40-344
IFCID 093 - IBM Service Record .............................................. 40-345
IFCID 094 - IBM Service Record .............................................. 40-346
IFCID 095 - Sort Start ....................................................... 40-347
IFCID 096 - Sort End ....................................................... 40-348
IFCID 097 - AMS Command End ............................................. 40-351
IFCID 098 - IBM Service Record .............................................. 40-352
IFCID 099 - IBM Service Record .............................................. 40-353
IFCID 100 - IBM Service Record ............................................. 40-354
IFCID 101 - IBM Service Record ............................................. 40-355
IFCID 102 - IBM Service Record ............................................. 40-356
IFCID 103 - SOS Off .......................................................... 40-357
IFCID 104 - Log Data Set ..................................................... 40-358
IFCID 105 - DBID/OBID Translation ........................................ 40-359
IFCID 106 - System Parameters .............................................. 40-360
  IFCID 106 - Application Programming Defaults ...................... 40-361
  IFCID 106 - Data Sharing Parameters .................................... 40-369
  IFCID 106 - Databases/Spaces Automatically Deferred ............... 40-372
  IFCID 106 - Databases/Spaces Automatically Restarted ............. 40-373
  IFCID 106 - Databases/Spaces Automatically Started ............... 40-374
  IFCID 106 - Distributed Data Facility Parameters .................... 40-375
  IFCID 106 - IRLM Processing Parameters ................................ 40-379
  IFCID 106 - Log Initialization Parameters (Part 1) ................ 40-382
  IFCID 106 - Log Initialization Parameters (Part 2) ................ 40-384
  IFCID 106 - Miscellaneous Installation Parameters .................. 40-389
  IFCID 106 - Stored Procedures Parameters ................................ 40-423
  IFCID 106 - System Initialization Parameters ......................... 40-424
  IFCID 106 - VSAM Catalog Name Qualifier .............................. 40-441
IFCID 107 - Open/Close ..................................................... 40-442
IFCID 108 - Bind Start ...................................................... 40-443
IFCID 109 - Bind End ...................................................... 40-448
IFCID 110 - Bind Free Start ................................................ 40-449
IFCID 111 - Bind Free End ................................................ 40-450
IFCID 112 - Thread Allocate ................................................ 40-451
IFCID 113 - Agent Allocate ................................................ 40-454
IFCID 114 - Archive Wait Start .......................................... 40-457
IFCID 115 - Archive Wait End DASD ........................................ 40-458
IFCID 116 - Archive Wait End Tape ........................................ 40-459
IFCID 117 - Archive Read Start .......................................... 40-460
IFCID 118 - Archive Read End ............................................ 40-461
IFCID 119 - BSDS Write Start ............................................. 40-462
IFCID 120 - BSDS Write End ............................................. 40-463
IFCID 121 - IBM Service Record ............................................ 40-464
| IFCID 122 | IBM Service Record | 40-465 |
| IFCID 123 | SRV Record | 40-466 |
| IFCID 124 | SQL Statement Record | 40-467 |
| IFCID 125 | RID Pool Processing | 40-470 |
| IFCID 126 | Log Buffer Write | 40-472 |
| IFCID 127 | Page Wait I/O In Prog (Start) | 40-473 |
| IFCID 128 | Page Wait I/O In Prog (End) | 40-475 |
| IFCID 129 | CI-S Obtained via IFI Reads | 40-477 |
| IFCID 140 | Audit Auth Failures | 40-478 |
| IFCID 141 | Audit DDL Grant/Revoke | 40-483 |
| IFCID 142 | Audit DDL Create/Alter/Drop | 40-487 |
| IFCID 143 | Audit First Write | 40-491 |
| IFCID 144 | Audit First Read | 40-492 |
| IFCID 145 | Audit DML Statement | 40-493 |
| IFCID 146 | User Record | 40-498 |
| IFCID 147 | Thread Summary | 40-499 |
| IFCID 147 - Data Sharing Accounting Data | 40-500 |
| IFCID 147 - Distributed Header Data | 40-501 |
| IFCID 147 - Instrumentation Accounting Data | 40-503 |
| IFCID 147 - Instrumentation Accounting Data Overflow | 40-504 |
| IFCID 147 - Logging | 40-507 |
| IFCID 147 - Monitor Detail Data | 40-508 |
| IFCID 147 - Thread Correlation Data | 40-514 |
| IFCID 149 | Resource Locking | 40-517 |
| IFCID 150 | Thread Locking | 40-518 |
| IFCID 150 - Global Interest Data | 40-519 |
| IFCID 150 - Held Lock Data | 40-521 |
| IFCID 150 - Lock Resource Data | 40-523 |
| IFCID 150 - Retained Lock Data | 40-525 |
| IFCID 150 - Suspend Lock Data | 40-526 |
| IFCID 151 | User Record | 40-528 |
| IFCID 152 | User Record | 40-529 |
| IFCID 153 | User Record | 40-530 |
| IFCID 154 | User Record | 40-531 |
| IFCID 155 | User Record | 40-532 |
| IFCID 156 | User Record | 40-533 |
| IFCID 157 | DRDS RDS Interface | 40-534 |
| IFCID 158 | DRDS CNV Interface | 40-535 |
| IFCID 159 | DRDS Req Site Data | 40-536 |
| IFCID 160 | DC Requester | 40-537 |
| IFCID 161 | DC Server | 40-538 |
| IFCID 162 | DTM Request | 40-539 |
| IFCID 163 | DTM Respond | 40-540 |
| IFCID 164 | IBM Service Record | 40-541 |
| IFCID 165 | IBM Service Record | 40-542 |
| IFCID 166 | IBM Service Record | 40-543 |
| IFCID 167 | Conv Alloc Req Queued | 40-544 |
| IFCID 168 | IBM Service Record | 40-545 |
| IFCID 169 | DIST Authid Translation | 40-546 |
| IFCID 170 | Suspend of Agent | 40-548 |
| IFCID 171 | IBM Service Record | 40-549 |
| IFCID 172 | Deadlock Data | 40-550 |
| IFCID 172 - Deadlock Header | 40-551 |
| IFCID 172 - Unit of Work - Resource | 40-552 |
| IFCID 172 - Unit of Work - Blocker | 40-553 |
| IFCID 172 - Unit of Work - Waiter | 40-556 |
| IFCID 173 | Class 2 Time | 40-560 |
| IFCID 174 | Arch Log CMD Sus Start | 40-561 |
| IFCID 175 | Arch Log CMD Sus End | 40-562 |
| IFCID 177 | Package Allocation | 40-563 |
| IFCID 178 | IBM Service Record | 40-566 |
IFCID 179 - IBM Service Record ................................ 40-567
IFCID 180 - DC Communication Buffers .......................... 40-568
IFCID 181 - IBM Service Record ................................ 40-570
IFCID 182 - IBM Service Record ................................ 40-571
IFCID 183 - DRDS RDS/SCC Interface ............................. 40-572
IFCID 184 - DC Communication Buffers .......................... 40-575
IFCID 185 - READs Data Capture Start ......................... 40-576
IFCID 186 - IBM Service Record ................................ 40-577
IFCID 188 - READs Data Capture End ............................ 40-578
IFCID 190 - IBM Service Record ................................ 40-579
IFCID 191 - DDM Level 6B Objects ................................. 40-581
  IFCID 191 - Command and/or Reply Section ..................... 40-582
  IFCID 191 - DB2 ZEDA ........................................ 40-584
  IFCID 191 - Header Section ................................... 40-585
  IFCID 191 - Late Descriptor Section ........................... 40-587
IFCID 191 - 6B DSS Section ...................................... 40-588
IFCID 192 - DDM Level 6A Header Errors .......................... 40-589
  IFCID 192 - Current 6A Header ................................ 40-590
  IFCID 192 - DDM Level 6A Header Errors ....................... 40-591
  IFCID 192 - Previous 6A Header ................................ 40-592
IFCID 193 - UOW/SQLCODE Mismatch ............................. 40-593
IFCID 194 - Invalid SNA FMH-5 Received .......................... 40-594
IFCID 195 - SQLDA Discrepancy ................................ 40-595
IFCID 196 - Timeout Data ....................................... 40-597
  IFCID 196 - Holder ........................................... 40-598
  IFCID 196 - Timeout Header ................................... 40-601
IFCID 197 - DB2 Messages ........................................ 40-603
IFCID 198 - Buffer Manager Page Access ......................... 40-604
IFCID 199 - Buffer Pool Statistics at Data Set Level .......... 40-606
IFCID 201 - Alter Buffer Pool ................................... 40-609
IFCID 202 - Buffer Pool Attributes ............................... 40-612
IFCID 203 - DDF Heuristic COMMIT/ROLLBK ....................... 40-615
IFCID 204 - DDF Partner Cold Start .............................. 40-617
IFCID 205 - DDF Warm Start Log Name Error Information ........ 40-619
  IFCID 205 - As Remembered by DB2 ........................... 40-620
  IFCID 205 - As Remembered by Partner ......................... 40-621
  IFCID 205 - DDF Warm Start Log Name Error ................... 40-622
IFCID 206 - DDF Protocol Error ................................ 40-623
IFCID 207 - DDF Heuristic Damage ................................ 40-625
IFCID 208 - DDF Syncpoint Protocol Error ....................... 40-627
IFCID 209 - DDF Syncpoint Comm Failure ......................... 40-629
IFCID 210 - Warm Start Log Name Change ......................... 40-631
IFCID 211 - Claim Data ......................................... 40-632
IFCID 212 - Drain Data ......................................... 40-633
IFCID 213 - Drain Lock Wait Start ............................... 40-635
IFCID 214 - Drain Lock Wait End ................................ 40-637
IFCID 215 - Claim Count 0 Wait Start ......................... 40-639
IFCID 216 - Claim Count 0 Wait End ............................. 40-640
IFCID 217 - Storage Pools ....................................... 40-641
  IFCID 217 - Additional Information ............................ 40-642
IFCID 217 - Agent Local Storage Pool Sizes ...................... 40-643
IFCID 217 - DBM1 Storage Pool Sizes ............................ 40-646
IFCID 217 - Storage Manager Pool Statistics .................... 40-648
IFCID 218 - Lock Avoidance Summary ............................ 40-651
IFCID 219 - Utility LISTDEF List Information .................... 40-652
IFCID 220 - Utility Data Set Information ......................... 40-653
IFCID 221 - Parallel Group Execution ............................ 40-655
  IFCID 221 - Buffer Pool Constrained Data (Section Type C) .. 40-656
IFCID 221 - Detail Buffer Pool Constrained Data (Section Type E) .. 40-657
IFCID 221 - Parallel Data ...................................... 40-658
IFCID 221 - Section Type D ..................................... 40-661
IFCID 222 - Parallel Group Elapsed Time ........................................... 40-662
IFCID 223 - Lock Avoidance Detail .................................................... 40-664
IFCID 224 - Select Procedure Bypassed ............................................. 40-665
IFCID 225 - System Storage Usage ...................................................... 40-666
  IFCID 225 - Address Space Summary - DBM1 .................................. 40-667
  IFCID 225 - Address Space Summary - DIST .................................. 40-668
  IFCID 225 - Statement Cache / XPROC Detail __________________________ 40-672
  IFCID 225 - Shared/Common Storage Summary __________________________ 40-675
  IFCID 225 - Storage Pool Details .................................................. 40-679
  IFCID 225 - Thread Information ................................................... 40-681
IFCID 226 - Page Latch Contention Start .......................................... 40-682
IFCID 227 - Page Latch Contention End ............................................ 40-684
IFCID 228 - Archive Deallocation Start ............................................ 40-685
IFCID 229 - Archive Deallocation End ............................................. 40-686
IFCID 230 - Group Buffer Pool Attributes ...................................... 40-687
IFCID 231 - Parallel Group Task Time ............................................. 40-690
IFCID 233 - Call User Routine ....................................................... 40-692
IFCID 234 - Calling Agent Auth IDs ................................................ 40-694
IFCID 236 - DDF SNA XLN Protocol Error ...................................... 40-695
IFCID 237 - Set Current Degree ..................................................... 40-696
IFCID 238 - IBM Service Record ..................................................... 40-697
IFCID 239 - Overflow Package/DBRM .............................................. 40-698
  IFCID 239 - Buffer Manager Accounting Data ................................ 40-699
  IFCID 239 - General Package Overflow Accounting Data ..................... 40-703
  IFCID 239 - Locking Data ................................................................ 40-704
  IFCID 239 - Package/DBRM Accounting Data ____________________________ 40-708
  IFCID 239 - RDS Package Accounting ........................................... 40-718
  IFCID 239 - Resource Limit Facility ............................................ 40-720
IFCID 247 - SQLDA Data and Input Host Variable Data ....................... 40-722
IFCID 248 - IBM Service Record ....................................................... 40-725
IFCID 249 - EDM Pool Invalidate DBD .............................................. 40-726
IFCID 250 - Connect/Rebuild Connect/Disconnect Group Bpool ............... 40-727
IFCID 251 - Buffer Manager PSET/Part P-Lock Request ....................... 40-730
IFCID 252 - IBM Service Record ..................................................... 40-732
IFCID 254 - Coupling Facility Cache Structure Statistics .................... 40-733
IFCID 255 - Buffer Refresh Due to XI .............................................. 40-736
IFCID 256 - Alter Group Buffer Pool ................................................ 40-738
IFCID 257 - IRLM Notify Req Detail ................................................ 40-740
IFCID 258 - Data Set Extend Activity .............................................. 40-743
IFCID 259 - Buffer Manager Pg P-Lock Req ...................................... 40-745
IFCID 260 - IBM Service Record ..................................................... 40-747
IFCID 261 - Group Buffer Pool Checkpoint ....................................... 40-748
  IFCID 262 - GBPOOLT Castout Threshold Processing .......................... 40-750
IFCID 263 - Page Set and Partition Castout Detail ................................ 40-752
IFCID 264 - IBM Service Record ..................................................... 40-755
IFCID 266 - IBM Service Record ..................................................... 40-756
IFCID 267 - CF Rebuild/Alter/Start .................................................. 40-757
IFCID 268 - CF Rebuild/Alter End .................................................... 40-758
IFCID 269 - Trusted/Context Trace .................................................. 40-761
IFCID 270 - Trusted/Context Trace .................................................. 40-764
IFCID 271 - Row Level and Column Level Access Control .................... 40-765
IFCID 272 - Associate Locators ....................................................... 40-767
IFCID 273 - Allocate Cursor ............................................................ 40-769
IFCID 305 - Table Check Constraint ................................................ 40-771
IFCID 311 - Global Temp Table Usage .............................................. 40-773
IFCID 313 - Uncommitted Unit of Recovery ..................................... 40-775
IFCID 314 - Authorization Exit Parameters ...................................... 40-777
IFCID 316 - SQL Statement Statistics ............................................. 40-779
IFCID 317 - SQL Statement String .................................................. 40-787
IFCID 319 - Audit Security Record .................................................. 40-788
IFCID 321 - Force-at-Commit Begin .................................................. 40-790
Chapter 41. The Record Trace File Data Set and Output Records

Part 8. The SQL Activity Report Set

Chapter 42. Introduction to the SQL Activity Report Set

Chapter 43. General SQL Activity Information

Summarization

Sorting

Workload Detail

Headers Used in SQL Activity

SQL Activity Report Header

SQL Activity Trace Header.
# SQL Activity Trace Index

## Chapter 44. The SQL Activity Report
- Examples of an SQL Activity Report .................................................. 44-2
- Example of an SQL Activity Report with Workload ................................. 44-5

## Chapter 45. The SQL Activity Trace
- Example of an SQL Activity Trace ....................................................... 45-2
- The SQL Activity Trace Index ............................................................ 45-7

## Chapter 46. SQL Activity Report and Trace Blocks
- SQL Detail Section .................................................................................. 46-2
- Report and Trace Details ....................................................................... 46-3
- Field Descriptions of SQL Activity Detail Report and Trace Details .......... 46-5
- Workload Detail ...................................................................................... 46-12
  - Accounting ......................................................................................... 46-13
  - Data Capture Activity ....................................................................... 46-14
  - Exit Activity ......................................................................................... 46-15
  - Function Resolution Activity .............................................................. 46-16
  - I/O Activity ......................................................................................... 46-18
  - Lock Suspension Activity ................................................................... 46-20
  - Minibind Activity ................................................................................ 46-22
  - Page and Row Locking Activity ........................................................... 46-29
  - Query Parallelism ................................................................................ 46-32
  - RID List Processing .............................................................................. 46-34
  - Scan Activity ....................................................................................... 46-35
  - Sort Activity ....................................................................................... 46-37
  - Sort Activity - QW0095/96 ................................................................. 46-38
  - Sort Activity - QW0028 ....................................................................... 46-41
  - Host Variables .................................................................................... 46-43
  - Workload Highlight ............................................................................ 46-46


## Chapter 47. Statistics Short Report ......................................................... 47-1

## Chapter 48. Statistics Long Report .......................................................... 48-1

## Chapter 49. Statistics Report and Trace Blocks ....................................... 49-1
- Accelerator ............................................................................................ 49-5
- Accounting Rollup ................................................................................ 49-9
- Aggregated Accounting Statistics .......................................................... 49-10
- Authorization Management .................................................................... 49-12
- Buffer Pool General .............................................................................. 49-15
- Buffer Pool Read ................................................................................... 49-20
- Buffer Pool Sort/Merge ......................................................................... 49-28
- Buffer Pool Write .................................................................................. 49-31
- Common Storage Below and Above 2 GB (DB2 10 or later) .................... 49-37
- CPU and Storage Metrics ....................................................................... 49-39
- CPU Times ........................................................................................... 49-41
- Data Capture .......................................................................................... 49-44
- Data Set Statistics .................................................................................. 49-45
- Data Sharing Locking ............................................................................ 49-47
- DBM1 and MVS Storage Below 2 GB .................................................... 49-52
- DBM1 Storage Above 2 GB .................................................................... 49-59
- Dynamic SQL Statement ....................................................................... 49-65
- DB2 API ................................................................................................. 49-68
- DB2 Commands ..................................................................................... 49-69
- DIST Storage Above 2 GB (DB2 10 or later) ............................................ 49-76
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>DIST and MVS Storage Below 2 GB (DB2 10 or later)</td>
<td>49-77</td>
</tr>
<tr>
<td>DRDA Remote Locations</td>
<td>49-80</td>
</tr>
<tr>
<td>EDM Pool Activity</td>
<td>49-86</td>
</tr>
<tr>
<td>EDM Pool Activity (DB2 9)</td>
<td>49-87</td>
</tr>
<tr>
<td>EDM Pool Activity (DB2 10 or later)</td>
<td>49-93</td>
</tr>
<tr>
<td>Global DDF Activity</td>
<td>49-98</td>
</tr>
<tr>
<td>Group Buffer Pool Activity</td>
<td>49-103</td>
</tr>
<tr>
<td>Highlights</td>
<td>49-115</td>
</tr>
<tr>
<td>IFC Destinations</td>
<td>49-121</td>
</tr>
<tr>
<td>IFC Record Counts</td>
<td>49-125</td>
</tr>
<tr>
<td>IRLM Storage Below and Above 2 GB (DB2 11)</td>
<td>49-127</td>
</tr>
<tr>
<td>Latch Counters</td>
<td>49-129</td>
</tr>
<tr>
<td>Locking Activity</td>
<td>49-133</td>
</tr>
<tr>
<td>Log Activity</td>
<td>49-137</td>
</tr>
<tr>
<td>Miscellaneous</td>
<td>49-142</td>
</tr>
<tr>
<td>MVS LPAR Shared Storage Above 2 GB (DB2 10 or later)</td>
<td>49-144</td>
</tr>
<tr>
<td>Open/Close Activity</td>
<td>49-146</td>
</tr>
<tr>
<td>Plan/Package Activity</td>
<td>49-148</td>
</tr>
<tr>
<td>Query Parallelism</td>
<td>49-153</td>
</tr>
<tr>
<td>Real and Auxiliary Storage for DBM1</td>
<td>49-157</td>
</tr>
<tr>
<td>Real and Auxiliary Storage for DIST (DB2 10 or later)</td>
<td>49-159</td>
</tr>
<tr>
<td>Real Storage in Use - Summary</td>
<td>49-161</td>
</tr>
<tr>
<td>RID List Processing</td>
<td>49-162</td>
</tr>
<tr>
<td>ROWID</td>
<td>49-166</td>
</tr>
<tr>
<td>Short-on-Storage Metrics</td>
<td>49-167</td>
</tr>
<tr>
<td>Stored Procedures</td>
<td>49-168</td>
</tr>
<tr>
<td>Subsystem Services</td>
<td>49-169</td>
</tr>
<tr>
<td>Subsystem Shared Storage Above 2 GB (DB2 10 or later)</td>
<td>49-175</td>
</tr>
<tr>
<td>SQL DCL</td>
<td>49-177</td>
</tr>
<tr>
<td>SQL DDL</td>
<td>49-179</td>
</tr>
<tr>
<td>SQL DML</td>
<td>49-185</td>
</tr>
<tr>
<td>Triggers</td>
<td>49-187</td>
</tr>
<tr>
<td>Use Currently Committed</td>
<td>49-188</td>
</tr>
<tr>
<td>User-Defined Functions</td>
<td>49-189</td>
</tr>
<tr>
<td>Workfile Database</td>
<td>49-190</td>
</tr>
</tbody>
</table>

Chapter 50. The Statistics Save-File Utility ........................................ 50-1

Chapter 51. The Statistics File Data Set and Output Records .................... 51-1

Part 10. The System Parameters Report Set

Chapter 52. System Parameters Report Header ........................................... 52-1

Chapter 53. Example of the System Parameters Report .................................. 53-1

Chapter 54. System Parameters Report Blocks ............................................ 54-1

Application Programming Defaults Panel 1 (DSNTIPF) ................................ 54-3
Application Programming Defaults Panel 2 (DSNTIP4, DSNTIP41) .................... 54-9
Archive Log Installation Parameters (DSNTIPA) ...................................... 54-13
Buffer Pool Parameters (DSNTIP1) ..................................................... 54-17
Data Definition Control Support (DSNTIPZ) ........................................... 54-19
Define Group or Member (DSNTIPK) ..................................................... 54-22
Databases and Spaces Started Automatically (DSNTIPSC) ........................... 54-24
Default Startup Modules (DSNTIP03) ................................................. 54-25
Distributed Data Facility Panel 1 (DSNTIPR) ........................................ 54-26
Chapter 55. Alter Buffer Pool Command Issued .............................................. 55-1
Chapter 56. Alter Group Buffer Pool Command Issued ................................. 56-1
Chapter 57. Buffer Pool Parameters ............................................................. 57-1
Chapter 58. Group Buffer Pool Parameters .................................................... 58-1

Part 11. The Utility Activity Report Set

Chapter 59. Headers Used in Utility Activity ................................................ 59-1
Utility Activity Report Header Example .......................................................... 59-3
Utility Activity Trace Header ............................................................... 59-4

Chapter 60. The Utility Activity Reports ....................................................... 60-1

Chapter 61. The Utility Activity Trace .......................................................... 61-1

Chapter 62. Workload Detail ..................................................................... 62-1
Bind Activity ................................................................................................. 62-2
Data Set Information ...................................................................................... 62-7
Exit Activity ..................................................................................................... 62-8
I/O Activity ....................................................................................................... 62-9
LISTDEF Information ..................................................................................... 62-11
Lock Suspension Activity ............................................................................ 62-12
Page and Row Locking Activity ................................................................. 62-14
Utility Phases ................................................................................................. 62-17
  Header Fields - Utility Phases ..................................................................... 62-19
  Field description - Utility phases ............................................................. 62-21

Part 12. Additional Information

Chapter 63. The DPMOUT Record ................................................................. 63-1

Chapter 64. OMEGAMON XE for DB2 PE VSAM Data Sets ......................... 64-1
About this publication

This publication provides a detailed description of each report.

This publication shows the reports produced by the following products:
- IBM® Tivoli® OMEGAMON® XE for DB2® Performance Expert on z/OS®
- IBM Tivoli OMEGAMON XE for DB2 Performance Monitor on z/OS

It gives examples of each report and describes the fields shown.

Note: In descriptions that apply to both, IBM Tivoli OMEGAMON XE for DB2 Performance Monitor on z/OS and IBM Tivoli OMEGAMON XE for DB2 Performance Expert on z/OS, the term OMEGAMON XE for DB2 PE is used for both.

Use this information to interpret OMEGAMON XE for DB2 PE reports. This information also supplies background and tuning information, where appropriate. If you need more conceptual information about OMEGAMON XE for DB2 PE reports and how reports are produced, refer to Reporting User’s Guide. For information about OMEGAMON XE for DB2 PE commands and command syntax, refer to Report Command Reference.

Always check the IBM DB2 and IMS™ Tools Library web page and the Tivoli library page for the most current version of this publication:
- Tivoli OMEGAMON XE for DB2 Performance Expert on z/OS (PDFs and Techdocs on DB2 Tools Product Page)
- Tivoli OMEGAMON XE for DB2 Performance Monitor on z/OS (PDFs and Techdocs on DB2 Tools Product Page)
- Tivoli Documentation Central

The technical changes for this edition are summarized under “What's new” on page xxxiii. Specific changes since the previous edition of this publication are indicated by a vertical bar (|) to the left of a change. Editorial changes that have no technical significance are not noted.

“Who should read this publication” on page xx
This information is for anyone who uses OMEGAMON XE for DB2 PE reports and traces to monitor DB2.

“Terminology used in this publication” on page xxii
This topic introduces the terminology used in this publication.

“Conventions used in the OMEGAMON documentation” on page xxiii
This information uses several conventions for special terms and actions, and operating system-dependent commands and paths.
Who should read this publication

This information is for anyone who uses OMEGAMON XE for DB2 PE reports and traces to monitor DB2.

You can monitor activities for:
• Determining DB2 system performance and efficiency
• Tuning DB2
• Identifying bottlenecks
• Measuring an application's performance and resource cost
• Evaluating the effects of application on other applications and the system
Terminology used in this publication

This topic introduces the terminology used in this publication.

Table 1 shows the terminology used in this publication.

Table 1. Terminology used in this publication

<table>
<thead>
<tr>
<th>Terms used in this information</th>
<th>... are abbreviated to or hereafter referred to as ...</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM Tivoli OMEGAMON XE for DB2 Performance Expert on z/OS</td>
<td>OMEGAMON XE for DB2 PE</td>
</tr>
<tr>
<td>IBM Tivoli OMEGAMON XE for DB2 Performance Monitor on z/OS</td>
<td>OMEGAMON XE for DB2 PM</td>
</tr>
<tr>
<td>IBM DB2 Buffer Pool Analyzer for z/OS</td>
<td>Buffer Pool Analyzer</td>
</tr>
</tbody>
</table>

Note: Performance Expert for Multiplatforms, Performance Expert for Workgroups, and Performance Expert for z/OS are abbreviated to OMEGAMON XE for DB2 PE where applicable.
Conventions used in the OMEGAMON documentation

This information uses several conventions for special terms and actions, and operating system-dependent commands and paths.

Panels and figures

The panels and figures in this document are representations. Actual product panels might differ.

Symbols

The following symbols might appear in command syntax:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Usage</th>
</tr>
</thead>
</table>
| | The or symbol is used to denote a choice. You can use the argument on the left or the argument on the right. For example:
| YES | NO |
| In this example, you can specify YES or NO. |
| () | Denotes optional arguments. Arguments that are not enclosed in square brackets are required. For example:
| APPLDEST DEST (ALTDEST) |
| In this example, DEST is a required argument and ALTDEST is optional. |
| {} | Some documents use braces to denote mandatory arguments, or to group arguments for clarity. For example:
| COMPARE {workload} - REPORT={SUMMARY | HISTOGRAM} |
| In this example, the workload variable is mandatory. The REPORT keyword must be specified with a value of SUMMARY or HISTOGRAM. |
| _ | Default values are underscored. For example:
| COPY infile outfile - [COMPRESS={YES | NO}] |
| In this example, the COMPRESS keyword is optional. If specified, the only valid values are YES or NO. If omitted, the default is YES. |

Notation conventions

The following conventions are used when referring to high-level qualifiers:

hilev A high-level qualifier. The high-level qualifier is the first prefix or set of prefixes in the data set name. Site-specific high-level qualifiers are shown in italics.

For example:
- thilev refers to the high-level qualifier for your target data set.
- rhilev refers to the high-level qualifier for your runtime data set.

For members in target libraries, the high-level qualifier is thilev rather than rhilev.

- shilev refers to the SMP/E library high-level qualifier.
Typeface conventions

This information uses the following typeface conventions:

**Bold**
- Interface controls (check boxes, push buttons, radio buttons, spin buttons, fields, folders, icons, list boxes, items inside list boxes, multicolumn lists, containers, menu choices, menu names, tabs, property sheets), labels (such as Note:)
- Keywords and parameters in text

**Italic**
- Words defined in text
- Emphasis of words (for example: Use the word that to introduce a restrictive clause.)
- New terms in text (except in a definition list)

**Monospaced**
- Examples and code examples
- File names, programming keywords, and other elements that are difficult to distinguish from surrounding text
- Message text and prompts addressed to the user
- Text that the user must type
- Values for arguments or command options

**Significant elements**

**Recommendation**
Provides guidance when more than one option is available.

**Related reading**
Refers you to other publications that contain relevant information.

**Requirement**
Identifies a condition that must be met to ensure that the product is functional.

**Restriction**
Identifies a restriction or limitation with this product or an associated procedure.

"Terminology used" on page xxv
IBM Tivoli OMEGAMON XE for DB2 Performance Monitor on z/OS can be considered as a functional subset of IBM Tivoli OMEGAMON XE for DB2 Performance Expert on z/OS. Therefore the abbreviation OMEGAMON XE for DB2 PE or DB2 PE is used for both products. If a distinction is required, OMEGAMON XE for DB2 PM or DB2 PM is used explicitly.

"How to read syntax diagrams" on page xxvi
The rules in this section apply to the syntax diagrams that are used in this publication.

"Where to find information" on page xxviii
You can access the documentation in several ways.

"Service updates and support information" on page xxix
You can access support information for IBM Tivoli OMEGAMON XE for DB2 Performance Expert on z/OS and IBM Tivoli OMEGAMON XE for DB2 Performance Monitor on z/OS on the Support home website, or you can use the IBM Support Assistant.
Accessibility features help people with a physical disability, such as restricted mobility or limited vision, or with other special needs, to use software products successfully. This information center is developed to comply with the accessibility requirements of software products according to Section 508 of the Rehabilitation Act of the United States.

Your feedback is important in helping to provide the most accurate and high-quality information.
Terminology used

IBM Tivoli OMEGAMON XE for DB2 Performance Monitor on z/OS can be considered as a functional subset of IBM Tivoli OMEGAMON XE for DB2 Performance Expert on z/OS. Therefore the abbreviation OMEGAMON XE for DB2 PE or DB2 PE is used for both products. If a distinction is required, OMEGAMON XE for DB2 PM or DB2 PM is used explicitly.

The following table shows the products that are described in this publication and the short names with which they are referred to throughout this publication:

<table>
<thead>
<tr>
<th>Product name</th>
<th>Short name</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM Tivoli OMEGAMON XE for DB2 Performance Expert on z/OS</td>
<td>OMEGAMON XE for DB2 PE or DB2 PE</td>
</tr>
<tr>
<td>IBM Tivoli OMEGAMON XE for DB2 Performance Monitor on z/OS</td>
<td>OMEGAMON XE for DB2 PM or DB2 PM</td>
</tr>
<tr>
<td>IBM DB2 Buffer Pool Analyzer for z/OS or a particular subsystem</td>
<td>Buffer Pool Analyzer</td>
</tr>
<tr>
<td>IBM DB2 database for z/OS</td>
<td>DB2</td>
</tr>
</tbody>
</table>

- Performance Expert Client and Workstation Online Monitor designate the client component of DB2 PE.
  The client component of DB2 PE also designates the end user interface of Performance Expert for Multiplatforms, Performance Expert for Workgroups, and DB2 PE.
- OMEGAMON Collector designates the server component of DB2 PE.
How to read syntax diagrams

The rules in this section apply to the syntax diagrams that are used in this publication.

Arrow symbols
Read the syntax diagrams from left to right, from top to bottom, following the path of the line.

- Two right arrows followed by a line indicate the beginning of a statement.
- One right arrow at the end of a line indicates that the statement syntax is continued on the next line.
- One right arrow followed by a line indicates that a statement is continued from the previous line.
- A line followed by a right arrow and a left error indicates the end of a statement.

Conventions
- SQL commands appear in uppercase.
- Variables appear in italics (for example, column-name). They represent user-defined parameters or suboptions.
- When entering commands, separate parameters and keywords by at least one blank if there is no intervening punctuation.
- Enter punctuation marks (slashes, commas, periods, parentheses, quotation marks, equal signs) and numbers exactly as given.
- Footnotes are shown by a number in parentheses, for example, (1).

Required items
Required items appear on the horizontal line (the main path).

Optional items
Optional items appear below the main path.

If an optional item appears above the main path, that item has no effect on the execution of the statement and is used only for readability.

Multiple required or optional items
If you can choose from two or more items, they appear vertically in a stack. If you must choose one of the items, one item of the stack appears on the stack main path.
If choosing one of the items is optional, the entire stack appears below the main path.

```
required-choice1
required-choice2
```

**Repeatable items**

An arrow returning to the left above the main line indicates that an item can be repeated.

```
REQUİRED-ITEM
repeatable-item
```

If the repeat arrow contains a comma, you must separate repeated items with a comma.

```
REQUİRED-ITEM
repeatable-item
```

If the repeat arrow contains a number in parenthesis, the number represents the maximum number of times that the item can be repeated.

```
REQUİRED-ITEM
(5)
repeatable-item
```

A repeat arrow above a stack indicates that you can specify more than one of the choices in the stack.

**Default keywords**

IBM-supplied default keywords appear above the main path, and the remaining choices are shown below the main path. In the parameter list following the syntax diagram, the default choices are underlined.

```
default-choice
required-choice1
required-choice2
```
Where to find information

You can access the documentation in several ways.

The documentation for this product is provided in PDF and in HTML format at the following websites:

- Tivoli OMEGAMON XE for DB2 Performance Expert on z/OS information center
- Tivoli OMEGAMON XE for DB2 Performance Monitor on z/OS information center

Accessing publications online

IBM posts publications for this and all other Tivoli products, as they become available and whenever they are updated, to the Tivoli software information center website. You can access the Tivoli software information center by going to the Tivoli Documentation Central website and clicking O under Tivoli Documentation A-Z to access all of the IBM Tivoli OMEGAMON product manuals.

Note: If you print PDF documents on other than letter-sized paper, set the option in the File > Print window that allows Adobe Reader to print letter-sized pages on your local paper.

The IBM Software Support website provides the latest information about known product limitations and workarounds in the form of technotes for your product. You can view this information at the Support home website.

Ordering publications

You can order many IBM publications such as product manuals or IBM Redbooks® online at the IBM Publications Center website.

You can also order by telephone by calling one of the following numbers:

- In the United States: 800-879-2755
- In Canada: 800-426-4968

In other countries, contact your software account representative to order Tivoli publications.

Accessing terminology online

The IBM Terminology website consolidates the terminology from IBM product libraries in one convenient location.
Service updates and support information

You can access support information for IBM Tivoli OMEGAMON XE for DB2 Performance Expert on z/OS and IBM Tivoli OMEGAMON XE for DB2 Performance Monitor on z/OS on the Support home website, or you can use the IBM Support Assistant.

Support home

On the Support home website, you can find service updates and support information including software fix packs, PTFs, Frequently Asked Questions (FAQs), technical notes, troubleshooting information, and downloads.

IBM Support assistant

The IBM Support Assistant (ISA) is a free tool that provides access to several IBM support resources in a single location. You can use the ISA tool to quickly access support-related information and serviceability tools for problem determination.

To use ISA, complete the following steps:
1. Download ISA from the IBM Software Support website.
2. Start the ISA tool.
   ISA runs as a web application in the default system-configured web browser.
3. Select the Updater tab.
4. Select the New Products and Tools tab.
   The plug-in features are categorized by product family.
5. Select Tivoli > IBM Tivoli OMEGAMON XE for DB2 Performance Expert on z/OS and IBM Tivoli OMEGAMON XE for DB2 Performance Monitor on z/OS.
6. Check the feature(s) to be installed and click Install.
7. Restart ISA.

To learn more about how to use ISA, click the Help link in the IBM Support Assistant window.
Accessibility features

Accessibility features help people with a physical disability, such as restricted mobility or limited vision, or with other special needs, to use software products successfully. This information center is developed to comply with the accessibility requirements of software products according to Section 508 of the Rehabilitation Act of the United States.

The accessibility features in this information center enable users to do the following tasks:

- Use assistive technologies, such as screen-reader software and digital speech synthesizer, to hear what is displayed on the screen. In this information center, all information is provided in HTML format. Consult the product documentation of the assistive technology for details on using assistive technologies with HTML-based information.
- Operate specific or equivalent features using only the keyboard.
- Magnify what is displayed on the screen.

In addition, all images are provided with alternative text so that users with vision impairments can understand the contents of the images.

Navigating the interface by using the keyboard

Standard shortcut and accelerator keys are used by the product and are documented by the operating system. Refer to the documentation provided by your operating system for more information.

Magnifying what is displayed on the screen

You can enlarge information in the product windows using facilities provided by the operating systems on which the product is run. For example, in a Microsoft Windows environment, you can lower the resolution of the screen to enlarge the font sizes of the text on the screen. Refer to the documentation provided by your operating system for more information.
How to send your comments

Your feedback is important in helping to provide the most accurate and high-quality information.

If you have any comments about this information or any other documentation, you can do one of the following actions:

- Complete and submit the Reader Comment Form.
- Send your comments by e-mail to swsdid@de.ibm.com.

Include the documentation name, the part number, the version number, and, if applicable, the specific location of the text you are commenting on (for example, a page number or table number).
What's new

This topic summarizes the significant improvements or enhancements for the product and refers you to the relevant topics for more information.

**SH12-6991-00 — October 2013**

This edition replaces IBM Tivoli OMEGAMON XE for DB2 Performance Expert on z/OS; IBM Tivoli OMEGAMON XE for DB2 Performance Monitor on z/OS: Report Reference, SH12-6963-00.

- Clarifications and corrections have been applied to the information where required.
- Existing block samples and descriptions are refreshed.
- OMEGAMON XE for DB2 PE version 5.2 supports DB2 9 or later. References to DB2 V8 have been removed.
- The DISTRIBUTE command has been removed from the information [Chapter 64, “OMEGAMON XE for DB2 PE VSAM Data Sets,” on page 64-1](#).
- The information in [Chapter 60, “The Utility Activity Reports,” on page 60-1](#), [“Header Fields - Utility Phases” on page 62-19](#) and [“Utility Phases” on page 62-17](#) has been updated.
- The samples have been updated for [Chapter 24, “EXPLAIN PACKAGE Command,” on page 24-1](#), [Chapter 25, “EXPLAIN SQLSTMT Command,” on page 25-1](#), and [Chapter 13, “Table PLAN_TABLE Data,” on page 13-1](#).
- The following IBM Service IFCIDs are inserted: IFCID 384, 385, 386, 397, 398 and 399
- The new long end-user fields for Accounting traces and reports improve the evaluation of the thread activity.
  - “MVS Accounting” on page 5-127
  - “Truncated Values” on page 5-225
  - “Accounting Report and Trace Blocks” on page 5-43
  - “Identification” on page 5-106
- New fields or report layouts are not listed in this summary.
- The field label and description of field QW0366FN is modified in “IFCID 366 - Incompatible Function Char(DEC) Executed” on page 40-828.
- The following blocks are removed from the information:
  - Work File Database (DB2 9)
  - IFCID 003 - Package/DBRM Accounting Data
  - IFCID 147 - Package/DBRM Accounting Data
- Working with Aggregated Accounting Statistics; the following report, trace, and record blocks have been added:
  - “Aggregated Accounting Statistics” on page 49-10
  - “IFCID 369 - Aggregated Accounting Statistics” on page 40-830
- Information for Accelerator data is added to:
  - “Converting Data Sets” on page 6-3
  - Chapter 7, “The Accounting File Data Set and Output Record,” on page 7-1
- “Converting Data Sets” on page 50-3
- Chapter 51, “The Statistics File Data Set and Output Records,” on page 51-1
Part 1. OMEGAMON XE for DB2 PE Logs

This topic provides information about the OMEGAMON XE for DB2 PE logs.

The OMEGAMON XE for DB2 PE logs provide summarized information about various events during OMEGAMON XE for DB2 PE execution. You can save some of this summarized information for use in later processing. The following events are reported:

- Records in exception status
- DB2 START/STOP TRACE commands
- Reduction interval completion by report set
- SAVE subcommand completion by report set
- RESTORE subcommand completion by report set
- Errors and messages
- IFCID record distribution

How to generate logs or how to prevent log generation

The OMEGAMON XE for DB2 PE logs are generated automatically for each OMEGAMON XE for DB2 PE execution, provided there are valid DD statements in your JCL. To prevent generation of these logs, omit the ddname from your JCL (the preferred method), or specify DUMMY in the definition.

Different log types

The following OMEGAMON XE for DB2 PE logs are available:

- The **DPMLOG execution log** provides a listing of messages issued during command stream validation and OMEGAMON XE for DB2 PE initialization. It also reports any errors during the execution of OMEGAMON XE for DB2 PE.
- The **exception log** provides a listing identifying accounting and statistics records with at least one field containing a value outside user-specified limits.
- The **job summary log** includes the following occurrences in OMEGAMON XE for DB2 PE processing:
  - Detection of a DB2 START TRACE or DB2 STOP TRACE command
  - Reduction interval completion by report set
  - SAVE subcommand completion by report set
  - RESTORE subcommand completion by report set
  - Key error and warning messages
- The **IFCID frequency distribution log** provides the count of the input and processed trace records accumulated by IFCID. For each IFCID, a percentage of the total number of input and processed records is calculated.

Chapter 1, “DPMLOG Execution Log,” on page 1-1
This topic provides details about the DPMLOG Execution log.

Chapter 2, “Exception Log,” on page 2-1
This topic provides details about the Exception log.

Chapter 3, “Job Summary Log,” on page 3-1
This topic provides details about the Job Summary Log.
Chapter 4, “IFCID Frequency Distribution Log,” on page 4-1

The IFCID Frequency Distribution log provides counts of the trace records by
IFCID. There are counts for the number of valid records provided as input to
OMEGAMON XE for DB2 PE as well as for the number of records that are
processed after GLOBAL filtering and after duplicate records are dropped.
Chapter 1. DPMLOG Execution Log

This topic provides details about the DPMLOG Execution log.

The DPMLOG Execution log shows:
• Messages issued during OMEGAMON XE for DB2 PE initialization
• Command stream syntax errors
• Information, warning, and error messages issued during processing

How to generate a DPMLOG Execution log

If the DPMLOG DD statement is omitted, such a statement is dynamically allocated and the output is directed to the default SYSOUT class specified for the job.

Example of a DPMLOG Execution log - SYSPRINT message log

The field labels shown in the following sample of a SYSPRINT message log are described in the following section.

<table>
<thead>
<tr>
<th>MSG.ID.</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>EPEC2001I</td>
<td>COMMAND INPUT FROM DDNAME SYSIN</td>
</tr>
<tr>
<td></td>
<td>ACCOUNTING</td>
</tr>
<tr>
<td></td>
<td>REDUCE</td>
</tr>
<tr>
<td></td>
<td>INTERVAL (5)</td>
</tr>
<tr>
<td></td>
<td>REPORT</td>
</tr>
<tr>
<td></td>
<td>ORDER (INTERVAL)</td>
</tr>
<tr>
<td>EXEC</td>
<td></td>
</tr>
<tr>
<td>EPEC1999I</td>
<td>SYSTEM INITIALIZATION COMPLETE. RETURN CODE 0</td>
</tr>
<tr>
<td>EPEC0999I</td>
<td>EXECUTION COMPLETE. RETURN CODE 0</td>
</tr>
</tbody>
</table>

The following sections describe the header and the fields in the DPMLOG Execution log.

“The DPMLOG Execution Log Header” on page 1-2
This topic describes the header of the DPMLOG Execution log.

“Field Descriptions” on page 1-3
This topic describes the fields shown in the DPMLOG Execution log.
The DPMLOG Execution Log Header

This topic describes the header of the DPMLOG Execution log.

The header of the DPMLOG Execution log contains the following information:

**OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (VnRnMn)**
- The product name and the version, release, and modification level.

**PAGE**
- The page number in the format *lll-nnnnnn*, where *lll* denotes the location number within the report and *nnnnnn* the page number within the location.

**EXECUTION LOG**
- The name of the log report.

**RUN DATE**
- The date and time of the OMEGAMON XE for DB2 PE job generating the log. The default format is *mm/dd/yy hh:mm:ss.th*, which can be changed with the DATEFORMAT parameter.
Field Descriptions

This topic describes the fields shown in the DPMLOG Execution log.

**MSG.ID.**

The message identification in the format *FPEcnnni*, where:

- *FPE* is the product code for OMEGAMON XE for DB2 PE
- *c* is the OMEGAMON XE for DB2 PE module component code
- *nnnn* is the error message number
- *i* is an action code. It can have the following values:
  - I (informational)
  - W (warning)
  - E (error)
  - S (severe error)
  - U (unrecoverable error)

**DESCRIPTION**

The complete text of the error message.
Chapter 2. Exception Log

This topic provides details about the Exception log.

The exception log identifies and lists Accounting and Statistics records with at least one field outside user-specified limits. You can use it to identify DB2 threads and Statistics intervals that contain fields with exceptional values. This helps you recognize performance problems in the DB2 subsystem and in threads.

Exception processing is accomplished by setting values in the exception threshold data set. You can define exception thresholds for specific Accounting and Statistics fields. When exception processing is requested, the instrumentation data is checked against these values. Only records with at least one field containing a value outside the user-specified limits are reported.

The exception log file data set is a sequential data set suitable for use by the DB2 load utility. It contains a listing of Accounting and Statistics exception records identical to the listing in the exception log.

Exception traces are available in the Accounting and Statistics report sets. Each of these relates separately to accounting or statistics data. The exception log reports Accounting and Statistics trace exceptions in the same report, in timestamp order. This helps you identify:

- Applications that might be causing exceptional conditions in the DB2 subsystem
- Exceptional DB2 subsystem conditions that might be causing thread performance problems

Although Accounting and Statistics exception reports are available in addition to traces, report entries are neither listed in the exception log nor stored in the exception log file data set.

Input to Exception Logs

DB2 Statistics and Accounting trace records with IFCID 001 and 002 (statistics) and IFCID 003 and 239 (accounting) are used as input to the exception log.
How to generate an Exception Log

This topic describes how to generate an Exception Log.

There is no OMEGAMON XE for DB2 PE command to generate the exception log. The exception log is generated automatically for an OMEGAMON XE for DB2 PE execution when the following DD statements are defined in your JCL:

**EXCPTDD**
Exception threshold data set

**EXTRCDD1**
Exception log

To prevent generation of the exception log, omit the EXTRCDD1 statement from your JCL (the preferred method), or specify DUMMY in the definition.

The amount of data reported in the exception log can be controlled by the GLOBAL INCLUDE or GLOBAL EXCLUDE and FROM and TO specifications.
Example of the Exception Log

This topic provides an example of an Exception Log.

Exception Log - example

The header and fields shown in the example of the Exception Log are described in the following sections.

The following sections describe the header and the fields in the Exception Log.

"The Exception Log Header" on page 2-4
This topic describes the header of the Exception log.

"Field Descriptions" on page 2-5
This topic describes the fields of an Exception Log.
The Exception Log Header

This topic describes the header of the Exception log.

The header of the Exception log contains the following information:

LOCATION
The DB2 reporting location. If the location name is not available, the DB2 data sharing group name is printed in this field. If the DB2 data sharing group name does not exist, the DB2 subsystem ID is printed.

GROUP
The name of the DB2 data sharing group. This field shows N/A if there is no group name.

MEMBER
The name of the DB2 data sharing member or the member name of the DB2 subsystem. This field shows N/A if there is no member name.

This field is not printed on the report page showing the GROUP BUFFERPOOLS PARAMETERS.

SUBSYSTEM
The ID of the DB2 subsystem that generated the data. This field is not printed on the report page showing the GROUP BUFFERPOOLS PARAMETERS.

DB2 VERSION
The DB2 version number of the subsystem that generated the data.

OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (VnRnMn)
The product name and the version, release, and modification level.

EXECUTION LOG
The name of the log report.

PAGE
The page number in the format lll-nnnnnn, where lll denotes the location number within the report and nnnnn the page number within the location.

ACTUAL FROM/TO
The date and time of the first and last record included in the log for a location, group, subsystem, or member.

PAGE DATE
The date of the timestamps printed on this page. A page break occurs at the change of the date. This is useful if a trace page contains more than one entry and the date is not shown for each entry.
Field Descriptions

This topic describes the fields of an Exception Log.

The following fields are shown including the OMEGAMON XE for DB2 PE identifiers.

**PRIMAUTH**
The primary authorization ID of the thread.

**ORIGAUTH**
The original authorization ID of the thread.

**PLANNNAME**
The DB2 application plan name of the thread.

**CONNECT**
The DB2 connection ID of the thread.

**CORRNAME**
The correlation name of the thread.

**CORRNMBR**
The correlation number of the thread.

**EXCEPTION TIME**
For accounting records, this is the accounting timestamp. For statistics records, this is the END TIME of the statistics interval in which the exception occurred.

**INSTANCE**
The LUW instance number.

**CONNTYPE**
The type of connection for the associated thread. Values are:
- **CICS**  CICS® Attach
- **DB2 PRIV**  DB2 private protocol
- **DB2CALL**  DB2 CALL Attach
- **DLI-BTCH**  DL/I Batch
- **DRDA**  DRDA® protocol
- **IMS-CNTL**  IMS Control Region
- **IMS-BMP**  IMS nontransaction-oriented BMP
- **IMS-MPP**  IMS Attach MPP
- **IMS-TBMP**  IMS transaction-oriented BMP
- **RRS**  RRS attach
- **TSO**  TSO foreground and background
Exception Log

**UTILITY**
Utility attach
If connection type is not present, ‘BLANK’ is printed.

**MAINPACK**
This identifier is used to distinguish plans according to the packages they contain.

**PER**
This identifies the log entry as an exception per system, per plan, or per program.

**FIELD ID**
The field ID of the accounting or statistics field in exception status.

**FIELD DESCRIPTION**
A description of the field in exception status. This description matches, as closely as possible, the terminology used in the Accounting and Statistics reports. If the field in exception status is a buffer pool field, the buffer pool ID is printed in front of the field description on the same line. Values are:
- BP0 — BP49
- BP32K — BP32K9

All nondistributed fields for an accounting thread or statistics interval are listed first. Any distributed fields in exception status follow the nondistributed fields and are grouped by remote location. Packages follow after DDF and are grouped by package name.

**BY**
The basis used for comparing values in the records to values in the exception threshold data set. Values are:
- TOTAL — an absolute value (the default)
- MINUTE — by minute
- SECOND — by second
- COMMIT — by commit
- THREAD — by thread

**CALCULATED OR FIELD VALUE**
The value from the field in exception status — either an absolute value or a value calculated according to the comparison basis.

Time values are reported in the format `ssssss.thtt`, where `ssssss` is time in seconds and `thtt` is in tenths, hundredths, thousandths, and ten-thousandths of seconds. Integer values such as aborts and selects are reported in the format `nnnnnnnnnnnn`. Other values are reported in the format `nnnnnnnn.nn`.

**OP**
The greater than (>) or less than (<) operator.

**THRESHOLD VALUE**
The value defined in the exception threshold data set, above or below which the actual value must fall to be considered in exception status.

**THRESHOLD TYPE**
Describes whether the THRESHOLD VALUE is defined in the exception threshold data set as a WARNING or a PROBLEM.

**Note:** PRIMAUTH, ORIGAUTH, PLANNNAME, CONNECT, CORRNAME, CORRNMGR, INSTANCE, CONNTYPE, and MAINPACK do not apply to statistics records. Except for MAINPACK, N/A is printed for these fields. For MAINPACK, nothing is printed.
Chapter 3. Job Summary Log

This topic provides details about the Job Summary Log.

The OMEGAMON XE for DB2 PE job summary log provides a summary of events during OMEGAMON XE for DB2 PE execution, and other information about DB2 that helps you interpret OMEGAMON XE for DB2 PE reports. The job summary log includes the following events:

- Detection of a DB2 START TRACE or DB2 STOP TRACE command.
- Reduction interval completion by report set.
  There is a summary of all intervals for each report set at the end of the reduction phase.
- RESTORE subcommand completion by report set. This includes the completion code, DB2 subsystem ID, timestamp information on any restored data, and the ddname of the RESTORE file.
- SAVE subcommand completion by report set. This includes the completion code, DB2 subsystem ID, timestamp information on any restored data, and the ddname of the SAVE file.

“How to Generate the Job Summary Log” on page 3-2
This topic shows how to generate a Job Summary log.

“Example of the Job Summary log” on page 3-3
This topic provides an example of the Job Summary log.

“Job Summary VSAM Data Set” on page 3-6
The job summary VSAM data set (JSSRSDD) is used for saving and restoring data-related Job Summary information.
How to Generate the Job Summary Log

This topic shows how to generate a Job Summary log.

There is no OMEGAMON XE for DB2 PE command to generate the job summary log. The log is generated automatically for each OMEGAMON XE for DB2 PE execution, provided that the appropriate ddname is defined in your JCL.

The ddname for the job summary log is JOBSUMDD.

To prevent generation of the job summary log, omit the ddname from your JCL (the preferred method), or specify DUMMY in the definition.

Note: Omitting the ddname for the job summary log also prevents the generation of the IFCID frequency distribution log because both reports are written to JOBSUMDD.
This topic provides an example of the Job Summary log.

**Job Summary log - example**

The fields shown in the example of the Job Summary log are described in the following sections.

**The Job Summary Log Header** on page 3-4

This topic describes the header of the Job Summary log.

**Field Descriptions** on page 3-5

This topic describes the fields of the Job Summary log.
The Job Summary Log Header

This topic describes the header of the Job Summary log.

The header of the Job Summary log contains the following information:

**OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (VnRnMn)**

The product name and the version, release, and modification level.

**PAGE**

The page number in the format **lll-nnnnnn**, where **lll** denotes the location number within the report and **nnnnnn** the page number within the location.

**JOB SUMMARY LOG**

The name of the log report.

**RUN DATE**

The date and time of the OMEGAMON XE for DB2 PE job generating the log. The default format is **mm/dd/yy hh:mm:ss.th**, which can be changed with the **DATEFORMAT** parameter.
Field Descriptions

This topic describes the fields of the Job Summary log.

The following fields are shown:

**MSG.ID.**
The message identification in the format FPEcnnnni, where:
- FPE is the product code for OMEGAMON XE for DB2 PE
- c is the OMEGAMON XE for DB2 PE module component code
- nnnn is the message number
- i is an action code with possible values of:
  - I (informational)
  - W (warning)
  - E (error)
  - S (severe error)
  - U (unrecoverable error)

**LOCATION**
The DB2 location to which the message applies. If there is no location data, the subsystem ID (DB2ID) is printed.

**DESCRIPTION**
The complete text of the message.

**GROUP**
The name of the data sharing group.

**SSID**
The ID of the data sharing subsystem.

**MEMBER**
The name of the data sharing member.

**TIME_STAMP**
The date and time of the current input trace record, in the format mm/dd/yy hh:mm:ss.th.
Job Summary VSAM Data Set

The job summary VSAM data set (JSSRSDD) is used for saving and restoring data-related Job Summary information.

When accounting or statistics data is saved and JSSRSDD has been included in the job stream, related Job Summary information is written to JSSRSDD. If JSSRSDD has been included in the job stream and data is restored, Job Summary information is restored to the job summary log.

If you are restoring data, the data set defined by JSSRSDD and the data set defined by the restore data set must match, that is, be produced by the same save operation.

JSSRSDD is optional. If you omit JSSRSDD, information about the previous processing of saved data is not restored or information about current processing is not saved.

The VSAM data set defined by JSSRSDD must already exist when you run OMEGAMON XE for DB2 PE. Either specify an existing data set from a previous OMEGAMON XE for DB2 PE run (when restoring data), or specify a new data set allocated using the IDCAMS DEFINE CLUSTER function. If an existing data set is used and the SAVE subcommand is specified, the new Job Summary data is added to the previous content.

See the Reporting User’s Guide for the attributes of OMEGAMON XE for DB2 PE VSAM data sets.

Note: Do not specify DUMMY for JSSRSDD.
Chapter 4. IFCID Frequency Distribution Log

The IFCID Frequency Distribution log provides counts of the trace records by IFCID. There are counts for the number of valid records provided as input to OMEGAMON XE for DB2 PE as well as for the number of records that are processed after GLOBAL filtering and after duplicate records are dropped.

An IFCID count is listed, and a percentage of the total number of records is calculated.

One copy of the IFCID Frequency Distribution log is produced for each location.

Input to the IFCID Frequency Distribution logs

All records supplied as input to OMEGAMON XE for DB2 PE are used automatically as input to the IFCID Frequency Distribution Log.

“How to Generate an IFCID Frequency Distribution Log” on page 4-2
This topic shows how to generate an IFCID Frequency Distribution log.

“Example of the IFCID Frequency Distribution Log” on page 4-3
This topic provides an example of an IFCID Frequency Distribution Log.
How to Generate an IFCID Frequency Distribution Log

This topic shows how to generate an IFCID Frequency Distribution log.

There is no OMEGAMON XE for DB2 PE command to generate the IFCID Frequency Distribution log. The log is generated automatically for each OMEGAMON XE for DB2 PE execution, provided that the appropriate ddname is defined in your JCL.

The ddname for the IFCID Frequency Distribution log is JOBSUMDD.

To prevent the generation of the IFCID Frequency Distribution log, omit the ddname from your JCL (the preferred method), or specify DUMMY in the definition.

Note: Omitting the ddname for the IFCID Frequency Distribution log also prevents the generation of the Job Summary log because both logs are written to JOBSUMDD.
Example of the IFCID Frequency Distribution Log

This topic provides an example of an IFCID Frequency Distribution Log.

IFCID Frequency Distribution Log - example

The header and fields shown in the example of the IFCID Frequency Distribution Log are described in the following sections.

<table>
<thead>
<tr>
<th>IFCID</th>
<th>INPUT COUNT</th>
<th>PCT OF TOTAL</th>
<th>INPUT COUNT</th>
<th>PCT OF TOTAL</th>
<th>INPUT COUNT</th>
<th>PCT OF TOTAL</th>
<th>INPUT COUNT</th>
<th>PCT OF TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>11</td>
<td>0.00%</td>
<td>11</td>
<td>5.23%</td>
<td>199</td>
<td>0.06%</td>
<td>143</td>
<td>68.09%</td>
</tr>
<tr>
<td>2</td>
<td>11</td>
<td>0.00%</td>
<td>11</td>
<td>5.23%</td>
<td>202</td>
<td>0.00%</td>
<td>11</td>
<td>5.23%</td>
</tr>
<tr>
<td>3</td>
<td>109,219</td>
<td>49.93%</td>
<td>0</td>
<td>0.00%</td>
<td>217</td>
<td>0.03%</td>
<td>0</td>
<td>0.00%</td>
</tr>
<tr>
<td>4</td>
<td>11</td>
<td>0.00%</td>
<td>0</td>
<td>0.00%</td>
<td>225</td>
<td>0.00%</td>
<td>11</td>
<td>5.23%</td>
</tr>
<tr>
<td>5</td>
<td>11</td>
<td>0.00%</td>
<td>0</td>
<td>0.00%</td>
<td>230</td>
<td>0.00%</td>
<td>11</td>
<td>5.23%</td>
</tr>
<tr>
<td>105</td>
<td>22</td>
<td>0.01%</td>
<td>0</td>
<td>0.00%</td>
<td>239</td>
<td>0.01%</td>
<td>109,170</td>
<td>49.91%</td>
</tr>
<tr>
<td>106</td>
<td>12</td>
<td>0.00%</td>
<td>12</td>
<td>5.71%</td>
<td>258</td>
<td>0.00%</td>
<td>0</td>
<td>0.00%</td>
</tr>
<tr>
<td>172</td>
<td>4</td>
<td>0.00%</td>
<td>0</td>
<td>0.00%</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

TOTAL INPUT TRACE RECORDS = 218,714
TOTAL PROCESSED TRACE RECORDS = 210

The following sections describe the header and the fields in the IFCID Frequency Distribution log.

"The IFCID Frequency Distribution Log Header" on page 4-4
This topic describes the header of the IFCID Frequency Distribution log.

"Field Descriptions" on page 4-5
This topic describes the fields of the IFCID Frequency Distribution log.
The IFCID Frequency Distribution Log Header

This topic describes the header of the IFCID Frequency Distribution log.

The header of the IFCID Frequency Distribution log contains the following information:

**LOCATION**
The DB2 reporting location. If the location name is not available, the DB2 data sharing group name is printed in this field. If the DB2 data sharing group name does not exist, the DB2 subsystem ID is printed.

**GROUP**
The name of the DB2 data sharing group. This field shows N/A if there is no group name.

**MEMBER**
The name of the DB2 data sharing member or the member name of the DB2 subsystem. This field shows N/A if there is no member name.

This field is not printed on the report page showing the GROUP BUFFERPOOLS PARAMETERS.

**SUBSYSTEM**
The ID of the DB2 subsystem that generated the data. This field is not printed on the report page showing the GROUP BUFFERPOOLS PARAMETERS.

**DB2 VERSION**
The DB2 version number of the subsystem that generated the data.

**OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (VnRnMn)**
The product name and the version, release, and modification level.

**IFCID FREQUENCY DISTRIBUTION LOG**
The title of the log report.

**PAGE**
The page number in the format lll-nnnnnn, where lll denotes the location number within the report and nnnnnn the page number within the location.

**RUN DATE**
The date and time of the OMEGAMON XE for DB2 PE job generating the log. The default format is mm/dd/yy hh:mm:ss.th, which can be changed with the DATEFORMAT parameter.

**ACTUAL FROM/TO**
The date and time of the first and last record included in the log for a location, group, subsystem, or member.
Field Descriptions

This topic describes the fields of the IFCID Frequency Distribution log.

The following fields are shown:

**IFCID**  The IFCID number of the record. The identifier is listed in decimal.

**INPUT COUNT**  The total number of occurrences of each IFCID in the raw data, after invalid records are rejected and partial GTF records are combined.

**INPUT PCT OF TOTAL**  The percentage of the total number of input records that the number in INPUT COUNT represents.

**PROCESSED COUNT**  The total number of occurrences of each IFCID in the processed data after GLOBAL filtering and after duplicate records are dropped. When DPMOUTDD is specified in the JCL, the value in this field is a reflection of the contents of the DPMOUT data set.

**PROCESSED PCT OF TOTAL**  The percentage of the total number of records in the processed data that the number in PROCESSED COUNT represents.

**TOTAL INPUT TRACE RECORDS**  The total of the INPUT COUNT column.

**TOTAL PROCESSED TRACE RECORDS**  The total of the PROCESSED COUNT column.
OMEGAMON XE for DB2 PE Logs
Part 2. The Accounting Report Set

These topics provide information about the Accounting reports.

Note: For an introduction to the Accounting report set and general accounting information refer to the Reporting User’s Guide. It also provides information on input to Accounting reports and traces.

Chapter 5, “Accounting Default Layouts,” on page 5-1
This topic provides examples of the Accounting default layout for SHORT and LONG.

Chapter 6, “The Accounting Save-File Utility,” on page 6-1
Use the Save-File utility to migrate and convert Accounting Save data sets into a format suitable for OMEGAMON XE for DB2 PE V5.2.

Chapter 7, “The Accounting File Data Set and Output Record,” on page 7-1
The FILE subcommand formats DB2 Accounting records and writes them to sequential data sets suitable for use by the DB2 load utility.
Accounting Report
Chapter 5. Accounting Default Layouts

This topic provides examples of the Accounting default layout for SHORT and LONG.

When data from a particular DB2 version is processed, N/A is printed for all fields in the report that are not applicable to that version.

For Accounting, the LAYOUT subcommand option ACCEL provides detailed thread-related Accelerator activity data.

You can use the user-tailored reporting (UTR) facility to modify the layouts and store the changes. If you do this, store your layouts under a different name to avoid confusion and keep the layouts relevant to this documentation.

Headers Used in Accounting” on page 5-2
This topic describes the header of the Accounting report layout.

How Averages Are Calculated” on page 5-4
Accounting reports show times and events averaged over the number of threads whilst Accounting traces show times and events as totals for each thread.

Accounting Report - Short” on page 5-6
The short Accounting report shows some of the most significant fields averaged over the number of threads.

Accounting Trace - Short” on page 5-20
The short Accounting trace shows some of the most significant fields summarized by thread.

Accounting Report - Long” on page 5-35
This topic shows an example of a long version of the Accounting report.

Accounting Trace - Long” on page 5-39
This topic shows an example of a long version of the Accounting trace.

Accounting Report and Trace Blocks” on page 5-43
Accounting reports and traces are arranged in blocks. Each block contains accounting information about a particular activity. The layout of each block is presented followed by the field descriptions.
Headers Used in Accounting

This topic describes the header of the Accounting report layout.

OMEGAMON XE for DB2 PE header information is printed at the top of each page of an Accounting report or trace. For a report, the header differs depending on whether it is a member-scope or group-scope report.

Accounting Report Header Member-Scope

Here is a sample header for an Accounting report generated with member-scope.

LOCATION: PMODBN1 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-1
GROUP: DBN1 ACCOUNTING REPORT - SHORT REQUESTED FROM: NOT SPECIFIED
MEMBER: SN11 TO: NOT SPECIFIED
SUBSYSTEM: SN11 ORDER: PRIMAUTH-PLANNAME INTERVAL FROM: 01/30/10 22:53:32.60
DB2 VERSION: V10 SCOPE: MEMBER TO: 01/30/10 22:50:05.07

Accounting Report Header-Group-Scope

Here is a sample header for an Accounting report generated with group-scope.

LOCATION: DSNCAT OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-1
GROUP: DSNCAT ACCOUNTING REPORT - SHORT REQUESTED FROM: NOT SPECIFIED
MEMBER: SN11 TO: NOT SPECIFIED
SUBSYSTEM: SN11 ORDER: PRIMAUTH-PLANNAME INTERVAL FROM: 01/30/10 18:47:13.28
DB2 VERSION: V10 PAGE DATE: 01/30/10

Accounting Trace Header

Here is a sample header for an Accounting trace.

LOCATION: PMODBN1 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-1
GROUP: DBN1 ACCOUNTING TRACE - SHORT REQUESTED FROM: NOT SPECIFIED
MEMBER: SN11 TO: NOT SPECIFIED
SUBSYSTEM: SN11 ACTUAL FROM: 01/30/10 22:53:32.60
DB2 VERSION: V10 PAGE DATE: 01/30/10

Description of the Accounting header fields

The Accounting headers, shown in “Accounting Report Header Member-Scope,” “Accounting Report Header-Group-Scope,” and “Accounting Trace Header,” contain the following information:

LOCATION
The DB2 reporting location. If the location name is not available, the DB2 data sharing group name is printed in this field. If the DB2 data sharing group name does not exist, the DB2 subsystem ID is printed.

GROUP
The name of the DB2 data sharing group. This field shows N/A if there is no group name.

MEMBER
The name of the DB2 data sharing member or the member name of the DB2 subsystem. This field shows N/A if there is no member name.

This field is not printed on the report page showing the GROUP BUFFERPOOLS PARAMETERS.

SUBSYSTEM
The ID of the DB2 subsystem that generated the data. This field is not printed on the report page showing the GROUP BUFFERPOOLS PARAMETERS.
DB2 VERSION
The DB2 version number of the subsystem that generated the data.

OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (VnRnMn)
The product name and the version, release, and modification level.

Title - layout
The title of the report and the layout. The layout can be a default layout provided with OMEGAMON XE for DB2 PE or a layout you have tailored yourself.

ORDER
If the ORDER option of the REPORT or TRACE subcommand was used to arrange the report entries, the selected keywords are shown in this field. Depending on the context, the OMEGAMON XE for DB2 PE identifiers by which lock events are grouped are shown here.

SCOPE
Scope of the report or trace, this can be MEMBER or GROUP. A member-scope report or trace shows data from a group for each individual member. In a group-scope report or trace, the data from individual members is consolidated and presented for the entire group.

PAGE
The page number in the format lll-nnnnnn, where lll denotes the location number within the report and nnnnn the page number within the location.

REQUESTED FROM and TO
The FROM and TO dates and times specified in the REPORT or TRACE subcommand.

If both FROM and TO dates and times are omitted from the REPORT subcommand, the FROM and TO dates and times specified in GLOBAL are printed. If only the FROM date and time or only the TO date and time has been specified, NOT SPECIFIED is printed for the unspecified value.

If FROM and TO are not specified in REPORT or GLOBAL, NOT SPECIFIED appears for both the FROM and TO values.

If you have specified FROM and TO times without dates in REPORT or GLOBAL, ALL DATES is printed along with the specified times.

INTERVAL FROM
The start date and time of the first reduction interval covered by the report. If REDUCE is not specified, the INTERVAL defaults to 0 and the timestamps of the first and last events are printed.

INTERVAL TO
The end date and time of the last reduction interval covered by the report. If REDUCE is not specified, the INTERVAL defaults to 0 and the timestamps of the first and last events are printed.

PAGE DATE
The date of the timestamps printed on this page. A page break occurs at the change of the date. This is useful if a trace page contains more than one entry and the date is not shown for each entry.
How Averages Are Calculated

Accounting reports show times and events averaged over the number of threads whilst Accounting traces show times and events as totals for each thread.

Fields in an Accounting report can show:

- **Averages** presented with two decimal places behind the point.
- **Totals** presented as whole numbers. If it is not possible to distinguish the type of data, totals are indicated with a hash (#) as the first character in the label.
- **Times** presented with six decimal places behind the point.

The calculation of package averages depends on the DB2 version installed:

- **Prior to DB2 10**: Averages are calculated by dividing totals by the number of occurrences of data that belongs to the counter.
  
The number of occurrences depends on the context of the data:
  
  - In general, thread level data is counted for **physical occurrences**.
  
  - If granularity is lost (because of rollup data), **logical occurrences** must also be taken into account. The logical number of occurrences on thread level is shown in the **#OCCURRENCES** field in the HIGHLIGHTS block of an (see "Highlights" on page 5-94) and in the field **#OCCURS** of the GENERAL block in an Accounting Short Report (see "Accounting Report - Short" on page 5-6).

  **Note**: If data sections were encountered, but the appropriate trace class was not active, invalid data would contribute to the result. Averages for Elapsed, CPU, and Suspension times would be affected in this case. For example, a time of zero would be added, although there is no real process that can be finished without consuming time. The average calculations are adjusted by this fact although these occurrences have not yet been reported.

  Averages of package counters are calculated by dividing the total value by ADTOTPOC, which is the number of package occurrences (sections) or DBRM:

  \[
  \text{Average} = \frac{\text{Total of package counter or time (QPACxxxx)}}{\text{Number of package sections processed (ADTOTPOC)}}
  \]

  **Note:**

  1. It is calculated regardless of enabled or disabled DB2 trace classes 7 and 8 at the time of writing the trace record for:

     - "Package Buffer Pool Activity - Class 10" on page 5-128
     - "Package Locking Activity - Class 10" on page 5-147
     - "Package SQL Activity - Class 10" on page 5-150

  2. It is calculated with respect to enabled DB2 trace class 7 at the time of writing the trace record. In query CP and sysplex query parallelism, these are aggregate records because parallel records are aggregated to the originating record. Together, they count as one record for:

     - "Package Times - Class 7" on page 5-160

  3. It is calculated with respect to enabled DB2 trace class 8 at the time of writing the trace record for:

     - "Package Times - Class 8 - Suspensions" on page 5-152

  4. For more information refer to the Accounting trace sections for a package.

- **DB2 10 or later**: Averages are calculated by dividing totals by QPACRLNU, which is the number of threads that roll data into this QPAC data section.

  \[
  \text{Average} = \frac{\text{Total of package counter or time (QPACxxxx)}}{\text{Number of threads that roll data into this QPAC data section (QPACRLNU)}}
  \]
This applies to package class 7, 8, or 10 times and events:
- “Package Buffer Pool Activity - Class 10” on page 5-128
- “Package Locking Activity - Class 10” on page 5-147
- “Package SQL Activity - Class 10” on page 5-150
- “Package Times - Class 7” on page 5-160
- “Package Global Contention P-Locks - Class 8” on page 5-134
- “Package Global Contention L-Locks - Class 8” on page 5-132
- “Package Times - Class 8 - Suspensions” on page 5-152
- “Package Identification - Report” on page 5-137
- “Package Identification - Trace” on page 5-142
- “Package General (Short Report)” on page 5-12
- “Package General (Short Trace)” on page 5-28

For more information refer to the Accounting trace sections for a package.

Averages of plan level counters in repeating data sections are calculated by dividing totals by the number of data sections which are aggregated to produce the report entry (this includes the DDF and RLF count).

Averages of buffer pool counters are calculated on a per-record basis.
Accounting Report - Short

The short Accounting report shows some of the most significant fields averaged over the number of threads.

Short Accounting reports are arranged in blocks. Each block contains Accounting information about a particular activity. The layout of each block is presented followed by the field descriptions.

You can generate a short version of the Accounting report using the following command:

```
... ACCOUNTING REPORT LAYOUT (SHORT) ORDER (PRIMAUTH-PLANNAME) SCOPE (MEMBER) ...
```

Accounting Report - Short

This is an example of a short Accounting report.

```
LOCATION: DODD911    OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-1
GROUP: N/P    ACCOUNTING REPORT - SHORT REQUESTED FROM: NOT SPECIFIED
MEMBER: N/P TO: NOT SPECIFIED
SUBSYSTEM: D911 ORDER: PRIMAUTH-PLANNAME INTERVAL FROM: 01/30/10 07:39:25.14
DB2 VERSION: V10 TO: 01/30/10 07:44:00.32

#OCCURS #COMMIT INSERTS OPENS PREPARE CLASS2 EL.TIME BUF.UPDT LOCK SUS
PRIMAUTH #DISTRS SELECTS UPDATES CLOSES CLASS1 EL.TIME CLASS2 CPUTIME SYN.READ #LOCKOUT
PLANNAME #ROLLBK FETCHES MERGES DELETES CLASS1 CPUTIME GETPAGES TOT.PREF

ABC 1164 1163 0.00 0.66 1.33 0.047610 198.13 0.00
java 1164 0.00 0.00 0.11 0.050089 0.009510 1.32 0

----------------------------------------------------------------------------------------------------------------
|PROGRAM NAME TYPE #OCCURS #ALLOCS SQLSTMT CL7 ELAP.TIME CL7 CPU TIME CL8 SUSP.TIME CL8 SUSP|
|PKGNAME PACKAGE 1164 1164 4.15 0.047610 0.009510 0.023908 4.47|
----------------------------------------------------------------------------------------------------------------

----------------------------------------------------------------------------------------------------------------
|REQUESTER METH #DDFS TRANS #ROLLBK #COMMIT SQLRECV ROWSENT CONVI|
|::FFFF:1.234.567 DRDA 1164 0.00 2 1163 2.78 64.40 0.00|
----------------------------------------------------------------------------------------------------------------

ACCOUNTING REPORT COMPLETE
```

“General (Short Report)” on page 5-7
This topic shows detailed information about “Accounting - General (Short Report)”.

“Package General (Short Report)” on page 5-12
This topic shows detailed information about “Accounting - Package General (Short Report)”.

“Distributed Activity Server (Short Report)” on page 5-15
This topic shows detailed information about “Accounting - Distributed Activity Server (Short Report)”.

“Distributed Activity Requester (Short Report)” on page 5-18
This topic shows detailed information about “Accounting - Distributed Activity Requester (Short Report)”.

5-6   OMEGAMON XE for DB2 PE & PM: Report Reference
General (Short Report)

This topic shows detailed information about “Accounting - General (Short Report)”.

This block is part of the Accounting Short Report.

Accounting - General (Short Report)

The field labels shown in the following sample layout of “Accounting - General (Short Report)” are described in the following section.

<table>
<thead>
<tr>
<th>#OCCURS</th>
<th>#COMMIT</th>
<th>INSERTS</th>
<th>OPENS</th>
<th>PREPARE</th>
<th>CLASS2</th>
<th>EL.TIME</th>
<th>BUF.UPDT</th>
<th>LOCK</th>
<th>SUS</th>
<th>PRIMAUTH</th>
<th>#DISTRS</th>
<th>SELECTS</th>
<th>UPDATES</th>
<th>CLOSES</th>
<th>CLASS1</th>
<th>EL.TIME</th>
<th>CLASS2</th>
<th>CPUTIME</th>
<th>SYN.READ</th>
<th>#LOCKOUT</th>
<th>PLANNAME</th>
<th>#ROLLBK</th>
<th>FETCHES</th>
<th>MERGES</th>
<th>DELETES</th>
<th>CLASS1</th>
<th>CPUTIME</th>
<th>GETPAGES</th>
<th>TOT.PREF</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABC</td>
<td>1164</td>
<td>1163</td>
<td>0.00</td>
<td>0.66</td>
<td>1.33</td>
<td>0.047610</td>
<td>198.13</td>
<td>0.00</td>
<td></td>
<td></td>
<td>1164</td>
<td>0.00</td>
<td>0.11</td>
<td>0.050089</td>
<td>327.38</td>
<td>11.30</td>
<td>0.00</td>
<td>198.13</td>
<td>0.00</td>
<td>0.047610</td>
<td>0.00</td>
<td>0.050089</td>
<td>327.38</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#OCCURS

The number of logical accounting records. A logical accounting record can contain more than one physical record.

This is the case, for example, in query CP and sysplex query parallelism, where several accounting records (IFCID 003 and, optionally, 239) are generated, namely one for the entire thread and one for each parallel task within the thread.

In case of Distributed Data Facility (DDF) or Recoverable Resource Manager Services Attach Facility (RRSAF) threads, it is the number of accounting intervals rolled up in a record.

This number is used for calculating averages (as a divisor) for class 1, 2, 3, and 5 times and events.

Field Name: ASOCCURS

#DISTRS

The number of accounting records with distributed activity. That is, the number of accounting records related to allied-distributed, DBAT, or DBAT-distributed threads.

Field Name: ASDISTRS

#ROLLBK

The number of rollback requests. This is the number of units that were backed out, including rollbacks from attaches.

Special Considerations: This field contains the number of:

- Application program abends
- Application rollback requests
- Application deadlocks on database records
- Applications canceled by operator
- Thread abends due to resource shortage

Field Name: QWACABRT

This is an exception field.
#COMMIT

The number of successful two-phase (units of recovery) or single-phase (syncs) commit requests. It indicates the number of units of recovery that are completed successfully, and for which the associated commit duration locks were released. It represents the total number of commit requests processed by the DB2 subsystem, whether the request was an explicit or implicit external request from an IMS or a CICS connection, or an implicit internal request within DB2 when DB2 was the commit coordinator or conducted read-only commit processing as a commit participant on phase-1 calls from an IMS or CICS connection.

For parallel queries, only the commits from the initiating (parent) thread are recorded by this counter.

**Field Name:** QWACCOMM

This is an *exception* field.

SELECTS

The number of SQL SELECT statements executed.

**Field Name:** QXSELECT

This is an *exception* field.

FETCHES

The number of FETCH statements executed. This number at the server location might not match the user application because of DDF's internal processing.

**Field Name:** QXFETCH

INSERTS

The number of INSERT statements executed.

**Field Name:** QXINSRT

UPDATES

The number of UPDATE statements executed.

**Field Name:** QXUPDTE

MERGES

The number of times a MERGE statement was executed.

**Field Name:** QXMERGE

OPENS

The number of OPEN statements executed.

**Field Name:** QXOPEN

CLOSES

The number of CLOSE statements executed. This number at the server location might not match the user application because of DDF's internal processing.

**Field Name:** QXCLOSE

DELETES

The number of DELETE statements executed.
Field Name: QXDELET

PREPARE

The number of SQL PREPARE statements executed. This number at the server location might not match the user application because of DDF's internal processing.

Field Name: QXPREP

CLASS1 EL.TIME

The class 1 elapsed time of the allied agent.

Special Considerations:
1. If the begin time equals zero, or if the end time minus begin time equals zero or is negative, N/C is shown.
2. Threads that can be reused, such as CICS protected threads or IMS/VS wait-for-input message regions, can include time during which the thread was inactive and waiting for work.
3. Elapsed time to process distributed requests is included for allied-distributed threads.
4. This time includes the time for processing SQL statements issued by stored procedures, user-defined functions, or triggers.
5. In query CP, sysplex query, or utility parallelism, this is the time shown in the originating record, which overlaps the elapsed times shown in the parallel records.

Field Name: ADRECETT

This is an exception field.

CLASS1 CPUTIME

The class 1 CPU time in an application. It indicates:
• The class 1 CPU time of the allied agent, which may include the accumulated class 1 TCB time for processing stored procedures, user-defined functions, and triggers.
• The accumulated CPU time for processing parallel tasks. This is valid for query CP parallelism, sysplex query parallelism, and parallel tasks generated by utilities.
• In sysplex query parallelism, the individual CPU times are normalized by the conversion factor of the parallel tasks that is related to the originating task.

In sysplex query parallelism, only CPU times of parallel tasks, running on the same member of the SYSPLEX group as the originating task, are included.

This CPU time does not include time that is consumed on an IBM specialty engine.

Field Name: ADCPUT

This is an exception field.

CLASS2 EL.TIME

The class 2 elapsed time of the allied agent accumulated in DB2.

Field Name: ADDB2ETT

This is an exception field.
CLASS2 CPUTIME

The class 2 CPU time (in DB2). It indicates:

- The class 2 CPU time for the allied agent. This includes the accumulated class 2 TCB time for processing any stored procedures, user-defined functions, and triggers.

- The accumulated CPU time for processing parallel tasks. This is valid for query CP parallelism, sysplex query parallelism, and parallel tasks generated by utilities.

- For batch reporting, in sysplex query parallelism, the individual CPU times are normalized by the conversion factor of the parallel tasks, related to the originating task.

For online monitoring, in sysplex query parallelism, only CPU times of parallel tasks, running on the same member of the sysplex group as the originating task, are included.

This CPU time does not include time that is consumed on an IBM specialty engine.

Field Name: ADDBCPUT

This is an exception field.

GETPAGES

The number of Getpage requests. This counter is incremented by successful Getpage requests for queries processed in parallel for each thread and for all successful and unsuccessful Getpage requests for queries that are not processed in parallel.

Background and Tuning Information

Reducing the number of Getpages can improve DB2 performance by reducing the number of synchronous page reads. With fewer Getpages, the requested page is more likely to be returned from the buffer pool. CPU usage is also reduced.

Check the ratio of Getpages to SQL DML statements, as a rule of thumb, try and keep this ratio below six.

You might need to modify the database and query design, for example:

- Add indexes to tables to reduce the number of pages scanned.
- Reassess the number of tables used and denormalize them, if necessary.
  As an example, a large table with many columns can result in several pages being fetched to satisfy a simple query requesting just a few columns. Splitting such a table into several tables with fewer columns, tailored to queries, will result in fewer pages returned for each query.
- Use correlated rather than noncorrelated queries to force the use of an index.

Field Name: QBACGET

This is an exception field.

BUF.UPDT

The number of times a buffer update occurs. This is incremented every time a page is updated and is ready to be written to DASD. If the same page is updated twice, for example, the number is incremented by 2.
This number is kept for all types of pages including data pages and work-file pages.

**Background and Tuning Information**

A nonzero value indicates any of the following activities:

- SQL INSERT, UPDATE, or DELETE
- Merge scan join
- Internal sort activity on the work files

Check the access path to determine whether sort activity can be minimized or avoided.

**Field Name:** QBACSWS

This is an *exception* field.

**SYN.READ**

The number of synchronous read I/O operations. DB2 increments this counter for each media manager synchronous physical read. Asynchronous I/O requests are not counted.

**Field Name:** QBACRIO

This is an *exception* field.

**TOT.PREF**

The number of sequential, dynamic, and list prefetch requests.

**Field Name:** ABCLSPR

This is an *exception* field.

**LOCK SUS**

The total number of all lock suspensions. This includes local and global lock suspensions.

**Field Name:** ALTSUSP

This is an *exception* field.

**#LOCKOUT**

The number of deadlocks and timeouts.

**Field Name:** ADTIMDLK

This is an *exception* field.
Package General (Short Report)

This topic shows detailed information about “Accounting - Package General (Short Report)”.

This block is part of the Accounting Short Report.

Accounting - Package General (Short Report)

The field labels shown in the following sample layout of “Accounting - Package General (Short Report)” are described in the following section.

<table>
<thead>
<tr>
<th>PROGRAM NAME</th>
<th>TYPE</th>
<th>#OCCURS</th>
<th>#ALLOC</th>
<th>SQLSTMT</th>
<th>CL7 ELAP.TIME</th>
<th>CL7 CPU TIME</th>
<th>CL8 SUSP.TIME</th>
<th>CL8 SUSP.</th>
</tr>
</thead>
<tbody>
<tr>
<td>PKGNAME</td>
<td>PACKAGE</td>
<td>1164</td>
<td>1164</td>
<td>4.15</td>
<td>0.047610</td>
<td>0.009510</td>
<td>0.023908</td>
<td>4.47</td>
</tr>
</tbody>
</table>

PROGRAM NAME

The program name (package ID or DBRM name).

In the case of rollup data (Accounting data of DDF/RRSAF threads and parallel tasks accumulated by DB2), the following value is shown:

- *ROLSUM* for DB2 10 or later
- *ROLLUP* for DB2 versions prior to DB2 10

Field Name: QPACPKID

This is an exception field.

TYPE

An indicator of whether the block describes a package or a DBRM. Possible values are PACKAGE, DBRM, and BOTH. BOTH can be shown in reports if there are packages and DBRMs with the same program name.

Field Name: ADPCKTYP

#OCCURS

This value can be one of the following:

- In general, the total number of accounting trace sections for a package or DBRM regardless of enabled or disabled DB2 trace classes 7 and 8 at the time of writing the trace record. In case of Distributed Data Facility (DDF) or Recoverable Resource Manager Services Attach Facility (RRSAF) threads, it is the number of accounting intervals rolled up in a record.

- If ORDER (ACTNAME) is specified, the total number of package sections of a special activity type depends on the following:
  - If IFCIDs 233 are available, stored procedures (SP) and user-defined functions (UDF) themselves are counted based on IFCID 233. Subprograms called by these routines and functions are not taken into account.
  - If IFCIDs 233 are not collected, all packages of an activity type are counted. The sum also includes the number of subprograms.

Field Name: ADTOTPOC

#ALLOCs

This value can be one of the following:
In general, the number of times a package was invoked by a different package. For the first package run by an application, the initial call counts as a package switch. If this package called a nested package (such as a trigger, UDF, or stored procedure), a switch will not be counted upon return from such a package.

- If ORDER (ACTNAME) is specified, the number of times a package of a special activity type is invoked from a different package depends on the following:
  - If IFCIDs 233 are available, the invocations of stored procedures (SP) and user-defined functions (UDF) themselves are counted based on IFCID 233. Invocations of subprograms called by these routines and functions are not taken into account.
  - If IFCIDs 233 are not collected, all invocations of an activity type are counted. The sum also includes the number of subprograms.

Field Name: APACSWIT

SQLSTMT

The number of SQL statements issued in this package or DBRM.
This number may not be equal to the total number of SQL statements in the QXST data section because QXST does not count all SQL statements. For example, it does not count commit or rollback statements.

Note: This field is shown for the following field labels in Accounting trace:
- SQL STMT - TOTAL
- SQL STMT - AVERAGE:
  - For DB2 9, the average is not calculated because it is identical to the TOTAL value. N/C (not calculated) is shown for this field.
  - For DB2 10 or later, the average data is shown.

Field Name: QPACSQLC

This is an exception field.

CL7 ELAP.TIME

The total elapsed time for executing the package or DBRM.

Field Name: QPACSCT

This is an exception field.

CL7 CPU TIME

The class 7 CPU time spent by the package or DBRM. It indicates:
- The TCB time
- The accumulated CPU time for processing parallel tasks. This is valid for query CP parallelism, sysplex query parallelism, and parallel tasks generated by utilities.

In sysplex query parallelism, only CPU times of parallel tasks, running on the same member of the sysplex group as the originating task, are included.

This time does not include the CPU time consumed on an IBM specialty engine.

Field Name: ADCPUTP
This is an *exception* field.

**CL8 SUSP:TIME**

The waiting time for the package or DBRM due to class 8 suspensions.

**Field Name:** ADTSUSTP

This is an *exception* field.

**CL8 SUSP**

The number of all types of class 8 suspensions.

**Field Name:** ADTSUSCP

This is an *exception* field.
Distributed Activity Server (Short Report)

This topic shows detailed information about “Accounting - Distributed Activity Server (Short Report)”.

This block is part of the Accounting Short Report.

Accounting - Distributed Activity Server (Short Report)

The field labels shown in the following sample layout of “Accounting - Distributed Activity Server (Short Report)” are described in the following section.

<table>
<thead>
<tr>
<th>SERVER</th>
<th>METH</th>
<th>#DDFS</th>
<th>TRANS</th>
<th>#ROLLBK</th>
<th>COMMIT</th>
<th>QLDSENT</th>
<th>ROWRECV</th>
<th>CONV1</th>
<th>CONVS</th>
<th>ELAPSED REQ</th>
<th>ELAPSED SER</th>
<th>SER</th>
<th>CPU</th>
</tr>
</thead>
<tbody>
<tr>
<td>123.45.678.90</td>
<td>DRDA</td>
<td>6</td>
<td>0.00</td>
<td>2</td>
<td>3</td>
<td>1.33</td>
<td>0.17</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>

...
Field Name: ADROL12S

#COMMIT
The total number of single-phase and two-phase commit requests sent.

Field Name: ADCOM12S

SQLSENT
The number of SQL statements sent to the server location. This value is maintained at the requesting location.

Field Name: QLACSQLS

ROWRECV
The number of rows of data retrieved from the server location. This value is maintained at the requester location.

Special Considerations:
1. The number of rows received from the server location does not include either the SQLDA or SQLCA.
2. Block fetch can significantly affect the number of rows sent across the network. When used with non-UPDATE cursors, block fetch puts as many rows as possible into the message buffer, and transmits the buffer across the network without requiring a VTAM® message. Consequently, more rows of data might be sent from the server location than are received by the reporting (requester) location. This is especially true when DB2 private protocol is used because multiple blocks can be transmitted from the server with no intervening messages sent by the requester.

Field Name: QLACROWR
This is an exception field.

CONVI
The number of conversations (both successful and unsuccessful) initiated by the requester location to be executed at the server location. This number is maintained at the requester.

Field Name: QLACCNVS
This is an exception field.

CONVS
The number of successful conversation allocations made to the server (DB2 private protocol only). This value is maintained at the requester location.

All allocation attempts, whether successful or not, are counted in QLACCNVS. The difference between QLACCNVS and this field helps to identify session resource constraint problems. Counting the number of unsuccessful conversations is useful for session tuning.

Field Name: QLACCNVA

ELAPSED REQ
The elapsed time at the requester. It includes the total of DB2 and network time.

Field Name: ADDSELRQ

ELAPSED SER
The elapsed database access agent time at the server location. This value is updated at the requester location.

Special Considerations:

- This value is reported only for DB2 private protocol. If only DRDA protocol, N/C is shown.
- If both DB2 private protocol and DRDA protocol are used, then only the elapsed time associated with the DB2 private protocol is reported, and this can be misleading.
- This value is calculated by accumulating the difference between the store clock values obtained after receiving a request message and before sending the associated reply message.
- When block fetch is used, this time can be longer than the time for ADDSELRQ (ELAPSED REQ).
- Compare this value with the accounting class 2 time (allied agent time in DB2) to see if the distributed-allied thread using the database access agent spends too much time in remote processing.

Field Name: ADDSELSR

This is an exception field.

SERVER CPU

The database access agent CPU time spent at the server location. This value is updated at the requester location, is intended for problem determination only, and should not be used for charge back.

Special Considerations:

1. This value is reported only for DB2 private protocol. If only DRDA protocol is used, N/C is shown.
2. If both DB2 private protocol and DRDA protocol are used, then only the CPU time associated with the DB2 private protocol is reported, and this can be misleading.
3. This value is calculated by accumulating the amount of CPU time spent by the database access thread at the DB2 server location each time a request message is processed.
4. Certain programming techniques can cause this value to not be received at the requester location (and therefore not included in this field), even though the CPU time was spent at the server location and was properly measured and sent to the requester location.

Field Name: ADDSSRSR

This is an exception field.
Distributed Activity Requester (Short Report)

This topic shows detailed information about “Accounting - Distributed Activity Requester (Short Report)”.

This block is part of the Accounting Short Report.

**Accounting - Distributed Activity Requester (Short Report)**

The field labels shown in the following sample layout of “Accounting - Distributed Activity Requester (Short Report)” are described in the following section.

| REQUESTER METH #DDFS TRANS #ROLLBK #COMMIT SQLRECV ROWSENT CONVI |
|-----------------------|---------------------|---------------------|---------------------|---------------------|
| ::FF::1.234.567 DRDA 1164 0.00 2 1163 2.78 64.40 0.00 |

**REQUESTER**

The name of the remote location with which this information is associated. If the local location is the requester, this field is a server location. If the local location is a server location, this field is the requester location. An allied thread is created at a DB2 requester, and a database access thread is created at a DB2 server. An accounting record is for either a requester or a server, but not for both.

This field is invalid in case of DB2 10 or later if summary rollup data is present. In Accounting this field is set to *ROLSUM*.

Field Name: QLACLOCN

This is an exception field.

**METH**

The method of access: DB2 private protocol, DRDA protocol, or both.

This field is invalid in case of DB2 10 or later if unique or summary rollup data is present. It can have the following value in:

- Accounting Trace and Report: N/P
- The Accounting FILE and SAVE PROGRAM table: blank

Field Name: ADPROTOC

**#DDFS**

The number of occurrences of the remote location and method pair.

Field Name: ASDDF

**TRANS**

The number of CREATE DATABASE ACCESS THREAD (DBAT) requests received by the server DBAT from the requester allied agent. This number is maintained by the server DBAT and is always 1.

Field Name: QLACTRNR

**#ROLLBK**

The total number of rollbacks (single phase and two-phase) received.

Field Name: ADROL12R

**#COMMIT**

The total number of commits (single phase and two-phase) received.
Field Name: ADCOM12R

SQLRECV

The number of SQL statements received from the requester location.

Field Name: QLACSQLR

ROWSENT

The number of rows sent from the server location to the requester location. The value includes SQLDA and is maintained at the server location.

Field Name: QLACROWS

CONVI

A count of conversations initiated by the requester. This number is updated at the server location.

Field Name: QLACCNVR
Accounting Trace - Short

The short Accounting trace shows some of the most significant fields summarized by thread.

Short Accounting traces are arranged in blocks. Each block contains Accounting information about a particular activity. The layout of each block is presented followed by the field descriptions.

The following example shows a short version of the Accounting trace produced by the following command:

```
... ACCOUNTING TRACE LAYOUT (SHORT) ...
```

**Accounting Trace - Short**

The following example shows a short version of the Accounting trace.

```
LOCATION: DODD911 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-1
GROUP: N/P ACCOUNTING TRACE - SHORT REQUESTED FROM: NOT SPECIFIED
MEMBER: N/P TO: NOT SPECIFIED
SUBSYSTEM: D911 ACTUAL FROM: 01/30/10 07:39:25.14
DB2 VERSION: V10 PAGE DATE: 01/30/10
```

```
PRIMAUTH CORRNAME ACCT TIMESTAMP SELECTS DELETES MERGES CPU TIME(CL1) GETPAGES TOT.PREF
PLANNAME CONNECT TERM. CONDITION OPENS INSERTS PREPARE EL. TIME(CL1) BUF.UPD'T LOCK SUS
NAME THRTYPE COMMITS FETCHES UPDATES EL. TIME(CL2) CPU TIME(CL2) SYN.READ LOCKOUTS
------- ------ ------------- --------------- ------- ------- ------------- ---------- --------
ABC 'BLANK' 07:39:25.143756 0 0 0 0.003189 42 1
java.exe SERVER NORM TYPE INACT 1 0 1 0.184705 0 0
java.exe DBAT 1 1 0 0.196676 0.003055 16 0
```

```
-----------------------------------------------------------------------------------------------
| PROGRAM NAME TYPE SQLSTMT CL7 ELAP.TIME CL7 CPU TIME CL8 SUSP.TIME CL8 SUSP |
| PKGNAME PACKAGE 4 0.184705 0.003055 0.041534 4 |
-----------------------------------------------------------------------------------------------
```

```
----------------------------------------------------------------------------
| REQUESTER METH TRANS ROLLBACK COMMITS SQLRECV ROWSENT CONVI |
| ::FFFF:9.123.456 DRDA10121 1 |
----------------------------------------------------------------------------
```

ACCOUNTING TRACE COMPLETE

“General (Short Trace)” on page 5-21
This topic shows detailed information about “Accounting - General (Short Trace)”.

“Package General (Short Trace)” on page 5-28
This topic shows detailed information about “Accounting - Package General (Short Trace)”.

“Distributed Activity Server (Short Trace)” on page 5-30
This topic shows detailed information about “Accounting - Distributed Activity Server (Short Trace)”.

“Distributed Activity Requester (Short Trace)” on page 5-33
This topic shows detailed information about “Accounting - Distributed Activity Requester (Short Trace)”.

5-20 OMEGAMON XE for DB2 PE & PM: Report Reference
General (Short Trace)

This topic shows detailed information about “Accounting - General (Short Trace)”. This block is part of the Accounting Short Trace.

Accounting - General (Short Trace)

The field labels shown in the following sample layout of “Accounting - General (Short Trace)” are described in the following section.

```
PRIMAUTH CORRNMBR ACCT TIMESTAMP SELECTS DELETES MERGES CPU TIME(CL1) GETPAGES TOT.PREF
PLANNAME CONNECT TERM. CONDITION OPENS INSERTS PREPARE EL. TIME(CL2) BUF.UPDT LOCK SUS
CORRNAME THR.TYPE COMMITS FETCHES UPDATES EL. TIME(CL1) CPU TIME(CL2) SYN.READ LOCKOUTS
---------- --------------- ------- ------- -------------- -------------- -------- --------
ABC "BLANK" 07:39:25.143756 0 0 0 0.003189 42 1
java.exe SERVER NORM TYP2 INACT 1 0 1 0.184705 0 0
java.exe OBAT 1 1 0 0.196676 0.003055 16 0
```

PRIMAUTH

The primary authorization ID from a connection or signon. The connection authorization exit and the signon authorization exit can change the primary authorization ID so that it differs from the original primary authorization ID (ORIGAUTH). Distributed authorization ID translation can also change the primary authorization ID.

Field Name: QWHCAID

PLANNAME

The plan name. It is blank for a DB2 command thread; otherwise:

**DSNESPRR**

For SPUFI with repeatable read.

**DSNESPCS**

For SPUFI with cursor stability.

**DSNUTIL**

For utilities.

**DSNTEP2**

For DSNTEP2.

**DSNBIND**

For binding.

The application plan name

For IMS.

The application plan name

For CICS.

A blank plan name

For IMS and CICS commands.

**DSQPLAN**

For QMF™.

The first 8 bytes of the application name

For DRDA connections to the common servers.

Field Name: QWHCPLAN

This is an exception field.
CORRNAME

This field shows the correlation name. It is obtained by translating the correlation ID into correlation name and number. The default translation depends on the connection type of the thread:

**Batch**  Job name

**TSO or CAF**  
Original authorization ID

**CICS**  Transaction ID

**IMS**  Application PST

**RRSAF**  
Characters 1 to 8 of the parameter correlation ID specified for SIGNON.

You can define your own correlation ID translation, which overrides the default translation.

**Field Name:** ADCORNME

CORRNMBR

This field shows the correlation number. It is obtained by translating the correlation ID into correlation name and number. The default translation depends on the connection type of the thread:

**Batch**  Blank

**TSO or CAF**  
Blank

**CICS**  Pool thread

**IMS**  Application PSBNAME

**RRSAF**  
Characters 9 - 12 of the parameter correlation ID specified for SIGNON.

You can define your own correlation ID translation which overrides the default translation.

**Field Name:** ADCORNMB

CONNECT

The connection name. Possible values are:

- For batch: BATCH
- For TSO: TSO
- For QMF: DB2CALL
- For utilities: UTILITY
- For DB2 private protocol this is the DB2 subsystem ID
- For IMS: the IMS ID
- For CICS, this is the CICS ID
- For DRDA connections from non-DB2 requesters: SERVER

**Field Name:** QWHCCN

This is an exception field.

THR.TYP

This is an exception field.
The type of thread. This field can contain one of the following values:

**ALLIED**
- The thread is not involved in any distributed activity.

**ALLDDIST**
- The thread is initiated by a DB2 attach and requests data from one or more server locations.

**DBAT**
- The thread is initiated, created, and performing work on behalf of a remote (requester) location. The value DBAT also includes DBAT DISTRIBUTED threads that are initiated by a requester location and executed by the server location that in turn requests data from another server location.

**Background and Tuning Information**

If the thread is involved in distributed activity, some monitored values can produce different results. For example, the class 1 elapsed time for a distributed thread is higher because VTAM time is also included.

**Field Name:** ADTHRTYP

**ACCT TIMESTAMP**
- The store clock value of the time when the accounting record was generated.

**Field Name:** QWHSSTCK

**TERM. CONDITION**
- The reason for termination, that is, for generating a DB2 accounting record.

**Field Name:** ADCNDRSN

**COMMITS**
- The number of successful two-phase (units of recovery) or single-phase (syncs) commit requests. It indicates the number of units of recovery that are completed successfully, and for which the associated commit duration locks were released. It represents the total number of commit requests processed by the DB2 subsystem, whether the request was an explicit or implicit external request from an IMS or a CICS connection, or an implicit internal request within DB2 when DB2 was the commit coordinator or conducted read-only commit processing as a commit participant on phase-1 calls from an IMS or CICS connection.

For parallel queries, only the commits from the initiating (parent) thread are recorded by this counter.

**Field Name:** QWACCOMM

This is an *exception* field.

**SELECTS**
- The number of SQL SELECT statements executed.

**Field Name:** QXSELECT

This is an *exception* field.

**OPENS**
- The number of OPEN statements executed.

**Field Name:** QXOPEN
This is an exception field.

**FETCHES**

The number of FETCH statements executed. This number at the server location might not match the user application because of DDF's internal processing.

*Field Name:* QXFETCH

**DELETES**

The number of DELETE statements executed.

*Field Name:* QXDELET

**INSERTS**

The number of INSERT statements executed.

*Field Name:* QXINSRT

**UPDATES**

The number of UPDATE statements executed.

*Field Name:* QXUPDTE

**MERGES**

The number of times a MERGE statement was executed.

*Field Name:* QXMERGE

**PREPARE**

The number of SQL PREPARE statements executed. This number at the server location might not match the user application because of DDF's internal processing.

*Field Name:* QXPREP

**EL. TIME(CL1)**

The class 1 elapsed time of the allied agent.

*Special Considerations:*

1. If the begin time equals zero, or if the end time minus begin time equals zero or is negative, N/C is shown.
2. Threads that can be reused, such as CICS protected threads or IMS/VS wait-for-input message regions, can include time during which the thread was inactive and waiting for work.
3. Elapsed time to process distributed requests is included for allied-distributed threads.
4. This time includes the time for processing SQL statements issued by stored procedures, user-defined functions, or triggers.
5. In query CP, sysplex query, or utility parallelism, this is the time shown in the originating record, which overlaps the elapsed times shown in the parallel records.

*Field Name:* ADRECETT

This is an exception field.

**CPU TIME(CL1)**

The class 1 CPU time in an application. It indicates:
• The class 1 CPU time of the allied agent, which may include the accumulated class 1 TCB time for processing stored procedures, user-defined functions, and triggers.

• The accumulated CPU time for processing parallel tasks. This is valid for query CP parallelism, sysplex query parallelism, and parallel tasks generated by utilities.

• In sysplex query parallelism, the individual CPU times are normalized by the conversion factor of the parallel tasks that is related to the originating task.
  
  In sysplex query parallelism, only CPU times of parallel tasks, running on the same member of the SYSPLEX group as the originating task, are included.

This CPU time does not include time that is consumed on an IBM specialty engine.

Field Name: ADCPUT

This is an exception field.

**EL. TIME(CL2)**

The class 2 elapsed time of the allied agent accumulated in DB2.

Field Name: ADDB2ETT

This is an exception field.

**CPU TIME(CL2)**

The class 2 CPU time (in DB2). It indicates:

• The class 2 CPU time for the allied agent. This includes the accumulated class 2 TCB time for processing any stored procedures, user-defined functions, and triggers.

• The accumulated CPU time for processing parallel tasks. This is valid for query CP parallelism, sysplex query parallelism, and parallel tasks generated by utilities.

• For batch reporting, in sysplex query parallelism, the individual CPU times are normalized by the conversion factor of the parallel tasks, related to the originating task.
  
  For online monitoring, in sysplex query parallelism, only CPU times of parallel tasks, running on the same member of the sysplex group as the originating task, are included.

This CPU time does not include time that is consumed on an IBM specialty engine.

Field Name: ADDBCPUT

This is an exception field.

**GETPAGES**

The number of Getpage requests. This counter is incremented by successful Getpage requests for queries processed in parallel for each thread and for all successful and unsuccessful Getpage requests for queries that are not processed in parallel.
Reducing the number of Getpages can improve DB2 performance by reducing the number of synchronous page reads. With fewer Getpages, the requested page is more likely to be returned from the buffer pool. CPU usage is also reduced.

Check the ratio of Getpages to SQL DML statements, as a rule of thumb, try and keep this ratio below six.

You might need to modify the database and query design, for example:
- Add indexes to tables to reduce the number of pages scanned.
- Reassess the number of tables used and denormalize them, if necessary.

As an example, a large table with many columns can result in several pages being fetched to satisfy a simple query requesting just a few columns. Splitting such a table into several tables with fewer columns, tailored to queries, will result in fewer pages returned for each query.
- Use correlated rather than noncorrelated queries to force the use of an index.

Field Name: QBACGET
This is an exception field.

BUF.UPDT
The number of times a buffer update occurs. This is incremented every time a page is updated and is ready to be written to DASD. If the same page is updated twice, for example, the number is incremented by 2.

This number is kept for all types of pages including data pages and work-file pages.

Background and Tuning Information
A nonzero value indicates any of the following activities:
- SQL INSERT, UPDATE, or DELETE
- Merge scan join
- Internal sort activity on the work files

Check the access path to determine whether sort activity can be minimized or avoided.

Field Name: QBACSWS
This is an exception field.

SYN.READ
The number of synchronous read I/O operations. DB2 increments this counter for each media manager synchronous physical read. Asynchronous I/O requests are not counted.

Field Name: QBACRIO
This is an exception field.

TOT.PREF
The number of sequential, dynamic, and list prefetch requests.

Field Name: ABCLSPR
This is an exception field.

LOCK SUS
The total number of all lock suspensions. This includes local and global lock suspensions.

**Field Name:** ALTSUSP

This is an *exception* field.

**LOCKOUTS**

The number of deadlocks and timeouts.

**Field Name:** ADTIMDLK

This is an *exception* field.
Package General (Short Trace)

This topic shows detailed information about “Accounting - Package General (Short Trace)”.

This block is part of the Accounting Short Trace.

Accounting - Package General (Short Trace)

The field labels shown in the following sample layout of “Accounting - Package General (Short Trace)” are described in the following section.

<table>
<thead>
<tr>
<th>PROGRAM NAME</th>
<th>TYPE</th>
<th>SQLSTMT</th>
<th>CL7 ELAP.TIME</th>
<th>CL7 CPU TIME</th>
<th>CL7 SUSP.TIME</th>
<th>CL8 SUSP TIME</th>
<th>CL8 SUSP</th>
</tr>
</thead>
<tbody>
<tr>
<td>PKGNAME</td>
<td>PACKAGE</td>
<td>4</td>
<td>0.184705</td>
<td>0.003055</td>
<td>0.041534</td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>

...%

PROGRAM NAME

The program name (package ID or DBRM name).

In the case of rollup data (Accounting data of DDF/RRSAF threads and parallel tasks accumulated by DB2), the following value is shown:

- *ROLSUM* for DB2 10 or later
- *ROLLUP* for DB2 versions prior to DB2 10

Field Name: QPACPKID

This is an exception field.

TYPE

An indicator of whether the block describes a package or a DBRM.

Possible values are PACKAGE, DBRM, and BOTH. BOTH can be shown in reports if there are packages and DBRMs with the same program name.

Field Name: ADPCKTYP

SQLSTMT

The number of SQL statements issued in this package or DBRM.

This number may not be equal to the total number of SQL statements in the QXST data section because QXST does not count all SQL statements. For example, it does not count commit or rollback statements.

Note: This field is shown for the following field labels in Accounting trace:

- SQL STMT - TOTAL
- SQL STMT - AVERAGE:
  - For DB2 9, the average is not calculated because it is identical to the TOTAL value. N/C (not calculated) is shown for this field.
  - For DB2 10 or later, the average data is shown.

Field Name: QPACSQLC

This is an exception field.

CL7 ELAP.TIME

The total elapsed time for executing the package or DBRM.

Field Name: QPACSCCT
This is an exception field.

**CL7 CPU TIME**

The class 7 CPU time spent by the package or DBRM. It indicates:

* The TCB time
* The accumulated CPU time for processing parallel tasks. This is valid for query CP parallelism, sysplex query parallelism, and parallel tasks generated by utilities.

In sysplex query parallelism, only CPU times of parallel tasks, running on the same member of the sysplex group as the originating task, are included.

This time does not include the CPU time consumed on an IBM specialty engine.

**Field Name:** ADCPUTP

This is an exception field.

**CL8 SUSP.TIME**

The waiting time for the package or DBRM due to class 8 suspensions.

**Field Name:** ADTSUSTP

This is an exception field.

**CL8 SUSP**

The number of all types of class 8 suspensions.

**Field Name:** ADTSUSCP

This is an exception field.
Distributed Activity Server (Short Trace)

This topic shows detailed information about “Accounting - Distributed Activity Server (Short Trace)”.

This block is part of the Accounting Short Trace.

Accounting - Distributed Activity Server (Short Trace)

The field labels shown in the following sample layout of “Accounting - Distributed Activity Server (Short Trace)” are described in the following section.

| SERVER METH TRANS ROLLBCK COMMITS SQLSENT ROWRECV CONVI CONVS CONVM ELAPSED REQ ELAPSED SER SERVER CPU |
| ::FFFF:1.234.567 DRDS 0.00 2 1163 2.78 64.40 0.00 0.00 0.00 0.001234 0.001345 0.001111 |

SERVER

The name of the remote location with which this information is associated. If the local location is the requester, this field is a server location. If the local location is a server location, this field is the requester location. An allied thread is created at a DB2 requester, and a database access thread is created at a DB2 server. An accounting record is for either a requester or a server, but not for both.

This field is invalid in case of DB2 10 or later if summary rollup data is present. In Accounting this field is set to *ROLSUM*.

Field Name: QLACLOCN

This is an exception field.

METH

The method of access: DB2 private protocol, DRDA protocol, or both.

This field is invalid in case of DB2 10 or later if unique or summary rollup data is present. It can have the following value in:

- Accounting Trace and Report: N/P
- The Accounting FILE and SAVE PROGRAM table: blank

Field Name: ADPROTOC

TRANS

The number of CREATE DATABASE ACCESS THREAD (DBAT) requests the requester allied agent sent to the server location. This number is maintained by the requester allied agent.

In some cases, for example when a new user signs on or a resignon occurs, the value of this field can be zero. This indicates that the existing DBAT at the server was reused by this user.

Field Name: QLACTRNS

ROLLBCK

The total number of rollbacks (single phase and two-phase) sent.

Field Name: ADROL12S

COMMITS

The total number of single-phase and two-phase commit requests sent.
Field Name: ADCOM12S

SQLSENT
The number of SQL statements sent to the server location. This value is maintained at the requesting location.

Field Name: QLACSQLS

ROWRECV
The number of rows of data retrieved from the server location. This value is maintained at the requester location.

Special Considerations:
1. The number of rows received from the server location does not include either the SQLDA or SQLCA.
2. Block fetch can significantly affect the number of rows sent across the network. When used with non-UPDATE cursors, block fetch puts as many rows as possible into the message buffer, and transmits the buffer across the network without requiring a VTAM message. Consequently, more rows of data might be sent from the server location than are received by the reporting (requester) location. This is especially true when DB2 private protocol is used because multiple blocks can be transmitted from the server with no intervening messages sent by the requester.

Field Name: QLACROWR
This is an exception field.

CONVI
The number of conversations (both successful and unsuccessful) initiated by the requester location to be executed at the server location. This number is maintained at the requester.

Field Name: QLACCNVS
This is an exception field.

CONVS
The number of successful conversation allocations made to the server (DB2 private protocol only). This value is maintained at the requester location.

All allocation attempts, whether successful or not, are counted in QLACCNVS. The difference between QLACCNVS and this field helps to identify session resource constraint problems. Counting the number of unsuccessful conversations is useful for session tuning.

Field Name: QLACCNVA

CONVM
The maximum number of conversations open at any time (QLACCNVA - QLACCNVT). QLACCIEL is updated only when (QLACCNVA - QLACCNVT) is greater than the current value of QLACCIEL. QLACFLG1 and QLACFLG2 indicate whether the conversations use DB2 private protocol, DRDA protocol, or both. This value is maintained at the requester location.

Field Name: QLACCIEL

ELAPSED REQ
The elapsed time at the requester. It includes the total of DB2 and network time.

Field Name: ADDSELRQ

ELAPSED SER

The elapsed database access agent time at the server location. This value is updated at the requester location.

Special Considerations:
- This value is reported only for DB2 private protocol. If only DRDA protocol, N/C is shown.
- If both DB2 private protocol and DRDA protocol are used, then only the elapsed time associated with the DB2 private protocol is reported, and this can be misleading.
- This value is calculated by accumulating the difference between the store clock values obtained after receiving a request message and before sending the associated reply message.
- When block fetch is used, this time can be longer than the time for ADDSELRQ (ELAPSED REQ).
- Compare this value with the accounting class 2 time (allied agent time in DB2) to see if the distributed-allied thread using the database access agent spends too much time in remote processing.

Field Name: ADDSELSR

This is an exception field.

SERVER CPU

The database access agent CPU time spent at the server location. This value is updated at the requester location, is intended for problem determination only, and should not be used for charge back.

Special Considerations:
1. This value is reported only for DB2 private protocol. If only DRDA protocol is used, N/C is shown.
2. If both DB2 private protocol and DRDA protocol are used, then only the CPU time associated with the DB2 private protocol is reported, and this can be misleading.
3. This value is calculated by accumulating the amount of CPU time spent by the database access thread at the DB2 server location each time a request message is processed.
4. Certain programming techniques can cause this value to not be received at the requester location (and therefore not included in this field), even though the CPU time was spent at the server location and was properly measured and sent to the requester location.

Field Name: ADDSSRSR

This is an exception field.
Distributed Activity Requester (Short Trace)

This topic shows detailed information about “Accounting - Distributed Activity Requester (Short Trace)”.

This block is part of the Accounting Short Trace.

Accounting - Distributed Activity Requester (Short Trace)

The field labels shown in the following sample layout of “Accounting - Distributed Activity Requester (Short Trace)” are described in the following section.

<table>
<thead>
<tr>
<th>REQUESTER</th>
<th>METH</th>
<th>TRANS</th>
<th>ROLLBCK</th>
<th>COMMITS</th>
<th>SQLRECV</th>
<th>ROWSENT</th>
<th>CONVI</th>
</tr>
</thead>
<tbody>
<tr>
<td>::FFFF:9.123.456</td>
<td>DRDA</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

REQUESTER

The name of the remote location with which this information is associated. If the local location is the requester, this field is a server location. If the local location is a server location, this field is the requester location. An allied thread is created at a DB2 requester, and a database access thread is created at a DB2 server. An accounting record is for either a requester or a server, but not for both.

This field is invalid in case of DB2 10 or later if summary rollup data is present. In Accounting this field is set to "ROLSUM".

Field Name: QLACLOCN

This is an exception field.

METH

The method of access: DB2 private protocol, DRDA protocol, or both.

This field is invalid in case of DB2 10 or later if unique or summary rollup data is present. It can have the following value in:

- Accounting Trace and Report: N/P
- The Accounting FILE and SAVE PROGRAM table: blank

Field Name: ADPROTOC

TRANS

The number of CREATE DATABASE ACCESS THREAD (DBAT) requests received by the server DBAT from the requester allied agent. This number is maintained by the server DBAT and is always 1.

Field Name: QLACTRNR

ROLLBCK

The total number of rollbacks (single phase and two-phase) received.

Field Name: ADROL12R

COMMITS

The total number of commits (single phase and two-phase) received.

Field Name: ADCOM12R

SQLRECV
The number of SQL statements received from the requester location.

Field Name: QLACSQLR

**ROWSENT**

The number of rows sent from the server location to the requester location. The value includes SQLDA and is maintained at the server location.

Field Name: QLACROWS

**CONVI**

A count of conversations initiated by the requester. This number is updated at the server location.

Field Name: QLACCNV
This topic shows an example of a long version of the Accounting report.

Use the following command to produce a long version of the Accounting report:

```
ACCOUNTING REPORT LAYOUT (LONG) ORDER (PRIMAUTH-PLANNAME) SCOPE (MEMBER)
```

### Accounting (Long Report)

<table>
<thead>
<tr>
<th>Location: PMODBE1</th>
<th>OMEGAMON XE for DB2 PERFORMANCE EXPERT (V5R2M0)</th>
<th>Page: 1-3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group: DBE1</td>
<td>Accounting Report - Long</td>
<td></td>
</tr>
<tr>
<td>Member: SE12</td>
<td>Requested from: Not Specified</td>
<td></td>
</tr>
<tr>
<td>Subsystem: SE12</td>
<td>Interval from: 02/18/13 15:34:00.00</td>
<td></td>
</tr>
<tr>
<td>DB2 Version: V11</td>
<td>To: 02/18/13 15:54:09.67</td>
<td></td>
</tr>
<tr>
<td>Primauth: KOZS</td>
<td>Planname: CLP</td>
<td></td>
</tr>
<tr>
<td>Member: SE12</td>
<td>Subsystem: SE12</td>
<td></td>
</tr>
<tr>
<td>Interval: 02/18/13</td>
<td>To: 02/18/13 15:54:09.67</td>
<td></td>
</tr>
</tbody>
</table>

#### Elapsed Time Distribution

<table>
<thead>
<tr>
<th>Category</th>
<th>Average Time</th>
<th>Class 2 Time Distribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPL</td>
<td>1.050354</td>
<td></td>
</tr>
<tr>
<td>CPU</td>
<td>0.038820</td>
<td></td>
</tr>
<tr>
<td>SUSP</td>
<td>0.000000</td>
<td></td>
</tr>
<tr>
<td>NOTACC</td>
<td>0.000000</td>
<td></td>
</tr>
<tr>
<td>SUSP</td>
<td>0.000000</td>
<td></td>
</tr>
</tbody>
</table>

#### Average Time

<table>
<thead>
<tr>
<th>Category</th>
<th>Average Time</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPL</td>
<td>1.050354</td>
<td></td>
</tr>
<tr>
<td>CPU</td>
<td>0.038820</td>
<td></td>
</tr>
<tr>
<td>SUSP</td>
<td>0.000000</td>
<td></td>
</tr>
<tr>
<td>NOTACC</td>
<td>0.000000</td>
<td></td>
</tr>
<tr>
<td>SUSP</td>
<td>0.000000</td>
<td></td>
</tr>
</tbody>
</table>

#### Global Contention

<table>
<thead>
<tr>
<th>Category</th>
<th>Average Time</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>L-LOCKS</td>
<td>0.000000</td>
<td></td>
</tr>
<tr>
<td>P-LOCKS</td>
<td>0.000000</td>
<td></td>
</tr>
</tbody>
</table>

#### SQL DML

<table>
<thead>
<tr>
<th>Category</th>
<th>Average Time</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>INSERT</td>
<td>0.000000</td>
<td></td>
</tr>
<tr>
<td>UPDATES</td>
<td>0.000000</td>
<td></td>
</tr>
<tr>
<td>ROWS</td>
<td>0.000000</td>
<td></td>
</tr>
</tbody>
</table>

Chapter 5. Accounting Default Layouts

5-35
<table>
<thead>
<tr>
<th>Command</th>
<th>Average</th>
<th>TOTAL</th>
<th>Command</th>
<th>Average</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>MERGE</td>
<td>0.00</td>
<td>0</td>
<td>SET CURR.DEGREE</td>
<td>0</td>
<td>TABLESPACE</td>
</tr>
<tr>
<td>DELETE</td>
<td>0.00</td>
<td>0</td>
<td>SET RULES</td>
<td>0</td>
<td>DATABASE</td>
</tr>
<tr>
<td>ROWS</td>
<td>0.00</td>
<td>0</td>
<td>SET CURR.PATH</td>
<td>0</td>
<td>STOGROUP</td>
</tr>
<tr>
<td>DESCRIBE</td>
<td>0.00</td>
<td>0</td>
<td>CONNECT TYPE 1</td>
<td>0</td>
<td>SYNONYM</td>
</tr>
<tr>
<td>PREPARE</td>
<td>0.21</td>
<td>5</td>
<td>SET CONNECTION</td>
<td>0</td>
<td>PACKAGE</td>
</tr>
<tr>
<td>OPEN</td>
<td>0.21</td>
<td>5</td>
<td>RELEASE</td>
<td>0</td>
<td>PROCEDURE</td>
</tr>
<tr>
<td>FETCH</td>
<td>0.00</td>
<td>0</td>
<td>CALL</td>
<td>0</td>
<td>FUNCTION</td>
</tr>
<tr>
<td>ROWS</td>
<td>1.25</td>
<td>30</td>
<td>ASSOC LOCATORS</td>
<td>0</td>
<td>TRIGGER</td>
</tr>
<tr>
<td>CLOSE</td>
<td>0.00</td>
<td>0</td>
<td>ALLOC CURSOR</td>
<td>0</td>
<td>DIST TYPE</td>
</tr>
<tr>
<td>DML-ALL</td>
<td>0.67</td>
<td>16</td>
<td>FREE LOCATOR</td>
<td>0</td>
<td>TRUST. CTX</td>
</tr>
<tr>
<td>DCL-ALL</td>
<td>0.17</td>
<td>4</td>
<td>HOLD LOCATOR</td>
<td>0</td>
<td>SEQUENCE</td>
</tr>
<tr>
<td>NORMAL TERM.</td>
<td></td>
<td></td>
<td>END USER</td>
<td>0.00</td>
<td>APPL.PROGR. ABEND</td>
</tr>
<tr>
<td>BLOCK STOR THR</td>
<td>0.00</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STALINESS THR</td>
<td>0.88</td>
<td>21</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Average</th>
<th>TOTAL</th>
<th>Command</th>
<th>Average</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>NS USER</td>
<td>0.00</td>
<td>0</td>
<td>APPL.PROGR. ABEND</td>
<td>0</td>
<td>APPL.PGM ABEND</td>
</tr>
<tr>
<td>DEALLOCATION</td>
<td>0.00</td>
<td>0</td>
<td>END OF MEMORY</td>
<td>0</td>
<td>END OF MEMORY</td>
</tr>
<tr>
<td>APPL.PROGR. END</td>
<td>0.00</td>
<td>0</td>
<td>RESOL.IN DB2</td>
<td>0</td>
<td>END OF TASK</td>
</tr>
<tr>
<td>RESIGNON</td>
<td>0.00</td>
<td>0</td>
<td>CANCEL FORCE</td>
<td>0</td>
<td>CANCEL FORCE</td>
</tr>
<tr>
<td>OBAT INACTIVE</td>
<td>0.00</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TYPE2 INACTIVE</td>
<td>0.00</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RRS COMMIT</td>
<td>0.00</td>
<td>0</td>
<td>END USER</td>
<td>0.00</td>
<td>APPL.PROGR. ABEND</td>
</tr>
<tr>
<td>END USER THRESH</td>
<td>0.00</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BLOCK STOR THR</td>
<td>0.00</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STALINESS THR</td>
<td>0.88</td>
<td>21</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Average</th>
<th>TOTAL</th>
<th>Command</th>
<th>Average</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>NS USER</td>
<td>0.00</td>
<td>0</td>
<td>APPL.PROGR. ABEND</td>
<td>0</td>
<td>APPL.PGM ABEND</td>
</tr>
<tr>
<td>DEALLOCATION</td>
<td>0.00</td>
<td>0</td>
<td>END OF MEMORY</td>
<td>0</td>
<td>END OF MEMORY</td>
</tr>
<tr>
<td>APPL.PROGR. END</td>
<td>0.00</td>
<td>0</td>
<td>RESOL.IN DB2</td>
<td>0</td>
<td>END OF TASK</td>
</tr>
<tr>
<td>RESIGNON</td>
<td>0.00</td>
<td>0</td>
<td>CANCEL FORCE</td>
<td>0</td>
<td>CANCEL FORCE</td>
</tr>
<tr>
<td>OBAT INACTIVE</td>
<td>0.00</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TYPE2 INACTIVE</td>
<td>0.00</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RRS COMMIT</td>
<td>0.00</td>
<td>0</td>
<td>END USER</td>
<td>0.00</td>
<td>APPL.PROGR. ABEND</td>
</tr>
<tr>
<td>END USER THRESH</td>
<td>0.00</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BLOCK STOR THR</td>
<td>0.00</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STALINESS THR</td>
<td>0.88</td>
<td>21</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Average</th>
<th>TOTAL</th>
<th>Command</th>
<th>Average</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>NEW USER</td>
<td>0.00</td>
<td>0</td>
<td>APPL.PROGR. ABEND</td>
<td>0</td>
<td>APPL.PGM ABEND</td>
</tr>
<tr>
<td>DEALLOCATION</td>
<td>0.00</td>
<td>0</td>
<td>END OF MEMORY</td>
<td>0</td>
<td>END OF MEMORY</td>
</tr>
<tr>
<td>APPL.PROGR. END</td>
<td>0.00</td>
<td>0</td>
<td>RESOL.IN DB2</td>
<td>0</td>
<td>END OF TASK</td>
</tr>
<tr>
<td>RESIGNON</td>
<td>0.00</td>
<td>0</td>
<td>CANCEL FORCE</td>
<td>0</td>
<td>CANCEL FORCE</td>
</tr>
<tr>
<td>OBAT INACTIVE</td>
<td>0.00</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TYPE2 INACTIVE</td>
<td>0.00</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RRS COMMIT</td>
<td>0.00</td>
<td>0</td>
<td>END USER</td>
<td>0.00</td>
<td>APPL.PROGR. ABEND</td>
</tr>
<tr>
<td>END USER THRESH</td>
<td>0.00</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BLOCK STOR THR</td>
<td>0.00</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STALINESS THR</td>
<td>0.88</td>
<td>21</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**Distributed Activity**

<table>
<thead>
<tr>
<th>REQUESTER</th>
<th>COMMIT(i) RECEIVED: 9</th>
<th>MESSAGES SENT: 4.14</th>
<th>RMS SENT: 1.43</th>
</tr>
</thead>
<tbody>
<tr>
<td>METHOD</td>
<td>DRODA PROTOCOL</td>
<td>SQL RECEIVED: 2.81</td>
<td>BYTES SENT: 1147.38</td>
</tr>
<tr>
<td>CONV.INITIATED</td>
<td>0.05</td>
<td>BYTES RECEIVED: 1036.71</td>
<td>#THREADS INDENT: 0</td>
</tr>
<tr>
<td>#COMMIT(2) RECEIVED: N/A</td>
<td>TRANSACTIONS RECEIVED: N/A</td>
<td>MSG.IN BUFFER: N/A</td>
<td></td>
</tr>
<tr>
<td>#BACKOUT(2) RECEIVED: N/A</td>
<td>COMMIT(2) RESSENT: N/A</td>
<td>LAST AGENT RECEIVED: N/A</td>
<td>FORGET SENT: N/A</td>
</tr>
<tr>
<td>#COMMIT(2) PERFORM.: N/A</td>
<td>BACKOUT(2) RESSENT: N/A</td>
<td>BACKOUT(2) PERFORM: N/A</td>
<td></td>
</tr>
</tbody>
</table>

---

**Chapter 5. Accounting Default Layouts**

---

**SQLCL2H20**

<table>
<thead>
<tr>
<th>VALUE</th>
<th>TIMES</th>
<th>SQLCL2H20</th>
<th>AVERAGE TIME</th>
<th>AVG.EV</th>
<th>TIME/EVENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>TYPE</td>
<td>PACKAGE</td>
<td>ELAP-CL7 TIME-AVG: 0.967171</td>
<td>LOCK/LATCH: 0.000000</td>
<td>0.000000</td>
<td>0.00</td>
</tr>
<tr>
<td>CP CPU TIME: 0.036885</td>
<td>IRU M LOCK/LATCH: 0.000000</td>
<td>0.000000</td>
<td>0.00</td>
<td>N/C</td>
<td></td>
</tr>
<tr>
<td>COLLECTION ID: NULL</td>
<td>PAR.TASKS: 0.000000</td>
<td>SYNCHRONOUS I/O: 0.029137</td>
<td>4.75</td>
<td>0.006134</td>
<td></td>
</tr>
<tr>
<td>PROGRAM NAME: SQLCL2H20</td>
<td>SE CPU: 0.000000</td>
<td>OTHER READ I/O: 0.000000</td>
<td>0.00</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>SUSPEND-CLB: 0.527201</td>
<td>OTHER WRITE I/O: 0.000000</td>
<td>0.000000</td>
<td>0.00</td>
<td>N/C</td>
<td></td>
</tr>
</tbody>
</table>

**Activity Type**

| NESTED | NONNESTED | AGENT | 0.527201 | SERV.TASK SWITCH: 0.426858 | 2.58 | 0.165235 |

---

**Performance Summary**

<table>
<thead>
<tr>
<th>CPU TIMES</th>
<th>SQLCL2H20</th>
<th>AVERAGE TIME</th>
<th>AVG.EV</th>
<th>TIME/EVENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>TYPE</td>
<td>PACKAGE</td>
<td>ELAP-CL7 TIME-AVG: 0.967171</td>
<td>LOCK/LATCH: 0.000000</td>
<td>0.000000</td>
</tr>
<tr>
<td>CP CPU TIME: 0.036885</td>
<td>IRU M LOCK/LATCH: 0.000000</td>
<td>0.000000</td>
<td>0.00</td>
<td>N/C</td>
</tr>
<tr>
<td>COLLECTION ID: NULL</td>
<td>PAR.TASKS: 0.000000</td>
<td>SYNCHRONOUS I/O: 0.029137</td>
<td>4.75</td>
<td>0.006134</td>
</tr>
<tr>
<td>PROGRAM NAME: SQLCL2H20</td>
<td>SE CPU: 0.000000</td>
<td>OTHER READ I/O: 0.000000</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>SUSPEND-CLB: 0.527201</td>
<td>OTHER WRITE I/O: 0.000000</td>
<td>0.000000</td>
<td>0.00</td>
<td>N/C</td>
</tr>
</tbody>
</table>

**Activity Type**

| NESTED | NONNESTED | AGENT | 0.527201 | SERV.TASK SWITCH: 0.426858 | 2.58 | 0.165235 |
### Database Activity Summary

**Activity Name:** BLANK

**Tasks:** 0.00

**ARCH.LOG:** QUIESCE 0.00

**SCHEMA Name:** BLANK

**Not Accounted:** 0.40

**Log Read:** 0.00

**Success Auth Check:** AVG.DB2 ENTRY/EXIT 1.71

**DB2 Entry/Exit:** 124

**Claim Release:** 0.00

**SQL STM - Average:** CP CPU SU 2099.75

**Agent:** 2099.75

**Global Contention:** 0.065

**NBR of ALLOCATIONS:** 24

**Page Latch:** 0.00

**SQL Stmt - Total:** SELECT CPU SU 2099.75

**CPU SU:** 0.00

**Notify Messages:** 0.00

**SQL Stmt - Average:** CPU SU 0.00

**CPU SU:** 1.29

**Global Contention:** 10.13

**SUCC AUTH CHECK:** DB2 ENTRY/EXIT 5.17

**Page Latch:** 0.00

**NBR OF ALLOCATIONS:** 24

**Drain Lock:** 0.00

**SQL Stmt - Average:** SE CPU SU 0.00

**CPU SU:** 0.00

**Accelerator:** 0.00

**SQL Stmt - Total:** PQ SYNCHRONIZATION 0.00

**CPU SU:** 0.00

**Total CLR Suspens.:** 0.02

**SQL Stmt - Average:** ACCELERATOR 0.00

**CPU SU:** 0.17

**Total CLR Suspens.:** 0.02

**SQL Stmt - Average:** ACCELERATOR 0.04

**CPU SU:** 4.25

**Total CLR Suspens.:** 0.02

---

### SQL Stmt - Average

<table>
<thead>
<tr>
<th>Statement Type</th>
<th>Average Time</th>
<th>Event Type</th>
<th>Total Time</th>
<th>Event Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECT</td>
<td>0.00</td>
<td>N/A</td>
<td>0</td>
<td>N/A</td>
</tr>
<tr>
<td>INSERT</td>
<td>0.21</td>
<td>5</td>
<td>5</td>
<td>0.21</td>
</tr>
<tr>
<td>UPDATE</td>
<td>0.04</td>
<td>1</td>
<td>0</td>
<td>0.04</td>
</tr>
<tr>
<td>DELETE</td>
<td>0.00</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>DESCRIBE</td>
<td>0.00</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>PREPARE</td>
<td>0.21</td>
<td>5</td>
<td>5</td>
<td>0.21</td>
</tr>
<tr>
<td>OPEN</td>
<td>0.21</td>
<td>5</td>
<td>5</td>
<td>0.21</td>
</tr>
<tr>
<td>FETCH</td>
<td>0.00</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>CLOSE</td>
<td>0.00</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>LOCK TABLE</td>
<td>0.00</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>CALL</td>
<td>0.00</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
</tr>
</tbody>
</table>

---

### SQL Stmt - Average

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Average</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetPAGES</td>
<td>36.63</td>
<td>879</td>
</tr>
<tr>
<td>Buffer Updates</td>
<td>11.08</td>
<td>266</td>
</tr>
<tr>
<td>Synchronous write</td>
<td>0.13</td>
<td>3</td>
</tr>
<tr>
<td>Synchronous read</td>
<td>2.58</td>
<td>62</td>
</tr>
<tr>
<td>SEQ. PREFETCH REQs</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>LIST PREFETCH REQs</td>
<td>0.08</td>
<td>2</td>
</tr>
<tr>
<td>DYN. PREFETCH REQs</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>Pages Read Asynch.</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>MAX PG/ROW LOCKS HELD</td>
<td>0.42</td>
<td>10</td>
</tr>
<tr>
<td>Lock Request</td>
<td>41.38</td>
<td>993</td>
</tr>
<tr>
<td>Unlock Request</td>
<td>9.75</td>
<td>234</td>
</tr>
<tr>
<td>Query Request</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>Change Request</td>
<td>1.71</td>
<td>41</td>
</tr>
<tr>
<td>Other Request</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>Total Suspensions</td>
<td>0.17</td>
<td>4</td>
</tr>
<tr>
<td>Lock Suspensions</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>IRLM Latch Suspens.</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>Other Suspens.</td>
<td>0.17</td>
<td>4</td>
</tr>
</tbody>
</table>

---

**Location:** PMDBE1

**Subsystem:** SE12

**Group:** DB1

**Member:** SE12

**Accounting Report - Long**

**Requested From:** NOT SPECIFIED

**Interval:** 02/18/13 15:34:00.00

**Scope:** USER

**Db2 Version:** V11

**Order:** PRIMAUTH-PLANNAME

**Page:** 1-4

**Omegamon XE for DB2 PE & PM: Report Reference**

---

**Accounting Report Complete**
Accounting Trace - Long

This topic shows an example of a long version of the Accounting trace.

The following example shows an extract from a long version of the Accounting trace produced by the following command:

```
ACCOUNTING
TRACE
LAYOUT (LONG)
```

Accounting (Long Trace)

The following example shows an extract from a long version of the Accounting trace.
<table>
<thead>
<tr>
<th>SQL DML TOTAL</th>
<th>SQL DCL TOTAL</th>
<th>SQ L DDL CREATE</th>
<th>DROP</th>
<th>ALTER</th>
<th>LOCKING</th>
<th>TOTAL</th>
<th>DATA SHARING</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECT</td>
<td>0</td>
<td>LOCK TABLE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>INSERT</td>
<td>0</td>
<td>CRT TTABLE</td>
<td>0</td>
<td>N/A</td>
<td>N/A</td>
<td>DEADLOCKS</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>RMD</td>
<td>0</td>
<td>DCL TTABLE</td>
<td>0</td>
<td>N/A</td>
<td>N/A</td>
<td>ESCAL,(SNAR)</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>UPDATE-STRTS</td>
<td>0</td>
<td>SET SOLID</td>
<td>0</td>
<td>N/A</td>
<td>N/A</td>
<td>ESCAL,(EXCL)</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>RMD</td>
<td>0</td>
<td>SET H_VAR.</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>MAX PG/RM LCK HELD 2</td>
<td>P-UNLOCK REQ</td>
<td>0</td>
</tr>
<tr>
<td>MERGE</td>
<td>0</td>
<td>SET DEGREE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>LOCK REQUEST</td>
<td>119</td>
<td>0</td>
</tr>
<tr>
<td>DELETE</td>
<td>0</td>
<td>SET RULES</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>UNLOCK REQUEST</td>
<td>66</td>
<td>0</td>
</tr>
<tr>
<td>DELETE</td>
<td>0</td>
<td>SET PATH</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>QUERY REQUEST</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SQL DML TOTAL</td>
<td></td>
<td>TOTAL</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>SQL DCL TOTAL</td>
<td></td>
<td>TOTAL</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>SQL DDL TOTAL</td>
<td></td>
<td>TOTAL</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>SQL DML CREATE</td>
<td></td>
<td>TOTAL</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>DROP</td>
<td></td>
<td>TOTAL</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>ALTER</td>
<td></td>
<td>TOTAL</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>LOCKING</td>
<td></td>
<td>TOTAL</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>TOTAL</td>
<td></td>
<td>TOTAL</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>DATA SHARING</td>
<td></td>
<td>TOTAL</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>TOTAL</td>
<td></td>
<td>TOTAL</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
</tbody>
</table>

**Additional tables and data not fully transcribed due to formatting issues.**
CSWL - MATCHES FOUND 0
CSWL - DUPLS CREATED 0

---- RESOURCE LIMIT FACILITY

TYPE: N/P TABLE ID: N/P SERV.UNITS: N/P CPU SECONDS: N/P MAX CPU SEC: N/P

<table>
<thead>
<tr>
<th>BP0</th>
<th>BPPOOL ACTIVITY</th>
<th>TOTAL</th>
<th>BP2</th>
<th>BPPOOL ACTIVITY</th>
<th>TOTAL</th>
<th>BP0</th>
<th>BPPOOL ACTIVITY</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>GETPAGES</td>
<td>156</td>
<td>GETPAGES</td>
<td>3</td>
<td>GETPAGES</td>
<td>159</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BUFFER UPDATES</td>
<td>45</td>
<td>BUFFER UPDATES</td>
<td>3</td>
<td>BUFFER UPDATES</td>
<td>48</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SYNCHRONOUS WRITE</td>
<td>0</td>
<td>SYNCHRONOUS WRITE</td>
<td>1</td>
<td>SYNCHRONOUS WRITE</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SYNCHRONOUS READ</td>
<td>0</td>
<td>SYNCHRONOUS READ</td>
<td>0</td>
<td>SYNCHRONOUS READ</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEQ. PREFETCH WRITTS</td>
<td>0</td>
<td>SEQ. PREFETCH WRITTS</td>
<td>0</td>
<td>SEQ. PREFETCH WRITTS</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LIST PREFETCH WRITTS</td>
<td>1</td>
<td>LIST PREFETCH WRITTS</td>
<td>0</td>
<td>LIST PREFETCH WRITTS</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DYN. PREFETCH WRITTS</td>
<td>0</td>
<td>DYN. PREFETCH WRITTS</td>
<td>0</td>
<td>DYN. PREFETCH WRITTS</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PAGES READ ASYNCHR.</td>
<td>0</td>
<td>PAGES READ ASYNCHR.</td>
<td>0</td>
<td>PAGES READ ASYNCHR.</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

GROUP BP0

<table>
<thead>
<tr>
<th>GROUP BP0</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>GBP-DEPEND GETPAGES</td>
<td>58</td>
</tr>
<tr>
<td>READ(XI)-DATA RETURN</td>
<td>0</td>
</tr>
<tr>
<td>READ(XI)-NO DATA RT</td>
<td>3</td>
</tr>
<tr>
<td>READ(NF)-DATA RETURN</td>
<td>0</td>
</tr>
<tr>
<td>READ(NF)-NO DATA RT</td>
<td>0</td>
</tr>
<tr>
<td>PREFETCH PAGES READ</td>
<td>0</td>
</tr>
<tr>
<td>CLEAN PAGES WRITTEN</td>
<td>0</td>
</tr>
<tr>
<td>UNREGISTER PAGE</td>
<td>0</td>
</tr>
<tr>
<td>ASYNCH GBP REQUESTS</td>
<td>9</td>
</tr>
<tr>
<td>ASYNCH SEC-GBP REQ</td>
<td>0</td>
</tr>
<tr>
<td>PG P-LOCK LOCK REQ</td>
<td>19</td>
</tr>
<tr>
<td>SPACE MAP PAGES</td>
<td>3</td>
</tr>
<tr>
<td>DATA PAGES</td>
<td>5</td>
</tr>
<tr>
<td>INDEX LEAF PAGES</td>
<td>11</td>
</tr>
<tr>
<td>PG P-LOCK UNLOCK REQ</td>
<td>19</td>
</tr>
<tr>
<td>PG P-LOCK LOCK SUSP</td>
<td>18</td>
</tr>
<tr>
<td>SPACE MAP PAGES</td>
<td>2</td>
</tr>
<tr>
<td>DATA PAGES</td>
<td>5</td>
</tr>
<tr>
<td>INDEX LEAF PAGES</td>
<td>11</td>
</tr>
<tr>
<td>WRITE AND REGISTER</td>
<td>15</td>
</tr>
<tr>
<td>WRITE &amp; REGISTER MULT</td>
<td>3</td>
</tr>
<tr>
<td>CHANGED PAGES WRITTEN</td>
<td>23</td>
</tr>
<tr>
<td>NO COMPL CHECKS SUSPEND</td>
<td>0</td>
</tr>
</tbody>
</table>

---- INITIAL DB2 COMMON SERVER OR UNIVERSAL JDBC DRIVER CORRELATION

PRODUCT ID: COMMON SERV
PRODUCT VERSION: V9 R7 M0
CLIENT PLATFORM: NT
CLIENT APPLNAME: CLP C:\PROGRAMDATA\I
CLIENT AUTHID: 'BLANK'
DDCS ACC.SUFFIX: 'BLANK'

---- DISTRIBUTED ACTIVITY

REQUESTER: ::FFFF:9.157.1#1 ROLLBACK(1) RECEIVED: 0 THREADS INDUBT: 0
PRODUCT ID: COMMON SERV
PRODUCT VERSION: V9 R7 M0
SQL RECEIVED: 3 ROWS SENT: 0
METHOD: DRDA PROTOCOL
MESSAGES RECEIVED: 5 CONVERSAT.INITIATED: 0
COMMTS(1) RECEIVED: 2 BYTES SENT: 217 NBR RLUP THREADS: 2
BYTES RECEIVED: 556
COMMIT(2) RECEIVED: N/A COMMIT(2) RESP.SENT: N/A PREPARE RECEIVED: N/A
BACKOUT(2) RECEIVED: N/A BACKOUT(2)RESP.SENT: N/A LAST AGENT RECV.: N/A
COMMIT(2) PERFORMED: N/A BACKOUT(2)PERFORMED: N/A MESSAGES IN BUFFER: N/A
TRANSACTIONS RECEIVED: N/A FORGET SENT: N/A

SQLC2H20 VALUE SQLC2H20 TIMES SQLC2H20 TIME EVENTS TIME/EVENT

Chapter 5. Accounting Default Layouts 5-41
<table>
<thead>
<tr>
<th>SQLC2H20</th>
<th>ELAPSED TIME</th>
<th>EVENTS</th>
<th>SQLC2H20</th>
<th>ELAPSED TIME</th>
<th>EVENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLOBAL CONTENTION L-LOCKS</td>
<td>0.113340</td>
<td>46</td>
<td>GLOBAL CONTENTION P-LOCKS</td>
<td>0.533559</td>
<td>66</td>
</tr>
<tr>
<td>PARENT (DB,TS,TAB, PART)</td>
<td>0.020525</td>
<td>0</td>
<td>PAGESET/PARTITION</td>
<td>0.000000</td>
<td>0</td>
</tr>
<tr>
<td>CHILD (PAGE, ROW)</td>
<td>0.008430</td>
<td>9</td>
<td>PAGE</td>
<td>0.124933</td>
<td>18</td>
</tr>
<tr>
<td>OTHER</td>
<td>0.082385</td>
<td>37</td>
<td>OTHER</td>
<td>0.388626</td>
<td>48</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SQLC2H20</th>
<th>TOTAL</th>
<th></th>
<th>SQLC2H20</th>
<th>TOTAL</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECT</td>
<td>0</td>
<td></td>
<td>INSERT</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>UPDATE</td>
<td>0</td>
<td></td>
<td>DELETE</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>DESCRIBE</td>
<td>0</td>
<td></td>
<td>PREPARE</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>OPEN</td>
<td>0</td>
<td></td>
<td>FETCH</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>CLOSE</td>
<td>0</td>
<td></td>
<td>LOCK TABLE</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>CALL</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| GLOBAL CONTENTION L-LOCKS | 0.113340     | 46     | GLOBAL CONTENTION P-LOCKS | 0.533559     | 66     |

**Location:** PMOQBE1
**Group:** DBE1
**Accounting Trace - Long**
**Member:** SE12
**Subsystem:** SE12

**Session Information:**
- **Begin Time:** 02/18/13 15:23:06.70
- **End Time:** N/P
- **Prod Typ:** COMMON
- **Prod Ver:** V9 R7 M0
- **Luw Net:** G9908776
- **Luw Lsn:** H5FD

**Plan Information:**
- **Plan Name:** CLP C:\P
- **Connect:** Server
- **Ws Name:** KOZGS_AV

**CMD:** CLP C:\PROGRAMDATA\IBM\DB2\DB2COPY1\DB2\TMP\CCSCRIPT130218182144741
Accounting reports and traces are arranged in blocks. Each block contains accounting information about a particular activity. The layout of each block is presented followed by the field descriptions.

The layout of the Accounting report blocks and the corresponding trace blocks is similar, the main difference is that Accounting reports show times and events averaged over the number of threads, and accounting traces show times and events as totals for each thread.

Fields in an Accounting report can show average values, totals or times. Normally the columns within the blocks of a report are labeled to indicate the type of data shown, and are shown as follows:

**Averages**
- Have two decimal places behind the point

**Totals**
- Are whole numbers

**Times**
- Have six decimal places behind the point

Where it is not possible to distinguish the type of data, totals are indicated with a hash (#) as the first character in the label.

This topic shows each block in alphabetical order. Each field in the block is listed in the order that it appears, showing the field name (as shown in the long report and trace) followed by a description.

Each block is presented in the default layout. Some blocks can have columns, rows or fields that are not included in the default layout. For example, the SQL DCL, SQL DML, RID List, buffer pool and group buffer pool activity blocks have a \_/COMMIT column that is not shown in the default layout. You can include columns, rows, and fields not shown in the default layouts with user-tailored reporting (UTR).

Field names used in short reports and traces can vary slightly from those used in the long versions. This is to allow the layout of the printed report or trace to align properly.

If a counter value or specific information in reports, in windows, or on panels is not shown, the following notation is used to indicate the reason:

- **N/A** Not applicable is shown if DB2 never produces a counter value in a specific context. Examples are:
  - A counter is not available in one DB2 version.
  - Counters are mutually exclusive.

- **N/C** Not calculated is shown for a derived field where the value cannot be calculated or is useless. Examples are:
  - A divide by zero (percentages, ratios).
  - Suppression of negative elapsed time values.
  - Required counter values for calculation marked as N/A or N/P.
  - Insufficient data or small counter values to allow significant statements (meaningless or misleading averages).
N/P  Not present is shown for a field where DB2 can present values, but does not in this instance. Examples are:

- When counter values are not generated because of operational conditions (a trace class is not active).
- An application does not provide a value because it is optional.

The following types of names or strings are used in this information:

**Short name or string**

A short name or short string is either the value of an original DB2 field if it is less than or equal to the defined length of the field, or it is the abbreviation of a longer value which is populated in a field of varying length.

**Unique name or string**

A unique name or unique string is a generated string based on the short string and its length, with a right-adjusted #-sign and a sequence number. This sequence number depends on the amount of long fields found during processing, which have the same string prefix and length as the short string. For example:

WSNAME: IS-255-012345678#1

**Long name or string**

A long name or long string is the complete string populated in a field of varying length. This depends on the context where it is used, for example, it can be up to 712 characters in length.

"Accelerator" on page 5-48
This topic shows detailed information about “Accounting - Accelerator”.

"Buffer Pool Activity" on page 5-52
This topic shows detailed information about “Accounting - Buffer Pool Activity”.

"Data Capture" on page 5-56
This topic shows detailed information about “Accounting - Data Capture”.

"Data Sharing Locking" on page 5-58
This topic shows detailed information about “Accounting - Data Sharing Locking”.

"Distributed Activity - Requester" on page 5-61
This topic shows detailed information about “Accounting - Distributed Activity - Requester”.

"Distributed Activity - Server" on page 5-70
This topic shows detailed information about “Accounting - Distributed Activity - Server”.

"Drain and Claim" on page 5-83
This topic shows detailed information about “Accounting - Drain and Claim”.

"Dynamic SQL Statement" on page 5-84
This topic shows detailed information about “Accounting - Dynamic SQL Statement”.

"Global Contention L-Locks" on page 5-86
This topic shows detailed information about “Accounting - Global Contention L-Locks”.

"Global Contention P-Locks" on page 5-88
This topic shows detailed information about “Accounting - Global Contention P-Locks”.
This topic shows detailed information about “Accounting - User-Defined Functions”.

“User-Defined Functions” on page 5-226
Accelerator

This topic shows detailed information about “Accounting - Accelerator”.

The Accounting Accelerator report block is shown for each accelerator that provided services to a DB2 thread. The block consists of three adjacent columns which contain the accelerator identification, the activity-related counters, and the corresponding times.

**Note:** The Accounting trace shows values and times for each Q8AC section. The Accounting report does not only show accumulated values and times, but also average values and times calculated for one occurrence. It shows the sum of a counter or time of all Q8AC sections processed, divided by the number of processed Q8AC sections.

For more information on the Accounting fields referred to in the field descriptions below, see:
- “[Times - Class 1 - Application Time” on page 5-195
- “[Times - Class 2 - DB2 Time” on page 5-202

In the following example both layouts are shown, the report layout is followed by the trace layout.

**Accounting - Accelerator**

The field labels shown in the following sample layout of “Accounting - Accelerator” are described in the following section.

---

### Report:

<table>
<thead>
<tr>
<th>ACCELERATOR IDENTIFIER</th>
<th>ACCELERATOR AVERAGE</th>
<th>ACCELERATOR TOTAL</th>
<th>ACCELERATOR IDENTIFIER</th>
<th>ACCELERATOR AVERAGE</th>
<th>ACCELERATOR TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRODUCT SERVER A0702010</td>
<td>CONNECTS 1.00 2</td>
<td>ELAPSED TIME 17.267459</td>
<td>PRODUCT SERVER S1MB405</td>
<td>OCCURRENCES 100.00 2</td>
<td>34.534918</td>
</tr>
<tr>
<td></td>
<td>REQUESTS 2.00 4</td>
<td>ACCUM ACCEL 0.000000</td>
<td></td>
<td>0.000000</td>
<td>0.000000</td>
</tr>
<tr>
<td></td>
<td>TIMED OUT 0.00 0</td>
<td>CPU TIME 0.004231</td>
<td></td>
<td>0.000000</td>
<td>0.008462</td>
</tr>
<tr>
<td></td>
<td>FAILED 0.00 0</td>
<td>SENT ACCUM ACCEL 0.000000</td>
<td></td>
<td>0.000000</td>
<td>0.000000</td>
</tr>
<tr>
<td></td>
<td>BYTES 1936.00 3872</td>
<td>RECEIVE ACCUM 0.000000</td>
<td></td>
<td>0.000000</td>
<td>0.000000</td>
</tr>
<tr>
<td></td>
<td>MESSAGES 22.00 22</td>
<td>WAIT TIME 0.000000</td>
<td></td>
<td>0.000000</td>
<td>0.000000</td>
</tr>
<tr>
<td></td>
<td>BLOCKS 0.00 0</td>
<td>BLOCKS 0.00 0</td>
<td></td>
<td>0.000000</td>
<td>0.000000</td>
</tr>
<tr>
<td></td>
<td>SEND 0.00 0</td>
<td>ROWS 0.00 0</td>
<td></td>
<td>0.000000</td>
<td>0.000000</td>
</tr>
</tbody>
</table>

---

### Trace:

<table>
<thead>
<tr>
<th>ACCELERATOR IDENTIFIER</th>
<th>ACCELERATOR TOTAL</th>
<th>ACCELERATOR IDENTIFIER</th>
<th>ACCELERATOR TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRODUCT SERVER &quot;BLANK&quot;</td>
<td>OCCURRENCES 1</td>
<td>ELAPSED TIME 19.598478</td>
<td>PRODUCT SERVER &quot;NULLS&quot;</td>
</tr>
<tr>
<td></td>
<td>CONNECTS 1</td>
<td>SVCS TCP/IP 19.598478</td>
<td></td>
</tr>
<tr>
<td></td>
<td>REQUESTS 2</td>
<td>ACCUM ACCEL 0.000000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TIMED OUT 0</td>
<td>CPU TIME 0.005270</td>
<td></td>
</tr>
<tr>
<td></td>
<td>FAILED 0</td>
<td>SENT ACCUM ACCEL 0.000000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>BYTES 1728</td>
<td>RECEIVE CLASS 1 0.000000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MESSAGES 17191</td>
<td>ELAPSED 27.721982</td>
<td></td>
</tr>
<tr>
<td></td>
<td>BLOCKS 12</td>
<td>CP CPU 0.017549</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ROWS 1</td>
<td>SE CPU 0.017549</td>
<td></td>
</tr>
</tbody>
</table>

---

PRODUCT
The accelerator product identifier.
Field Name: Q8ACPRID

SERVER
The accelerator server identifier.
Field Name: Q8ACNAME

OCCURRENCES
The number of sections processed for the accelerator. The name of this accelerator is shown in the report in block ACCELERATOR IDENTIFIER.
Field Name: AIOCCUR

CONNECTS
The number of accelerator connects.
Field Name: Q8ACCONN

REQUESTS
The number of accelerator requests.
Field Name: Q8ACREQ

TIMED OUT
The number of timed out requests.
Field Name: Q8ACTOUT

FAILED
The number of failed requests.
Field Name: Q8ACFAIL

SENT - BYTES
The number of bytes sent.
Field Name: Q8ACBYTES

SENT - MESSAGES
The number of messages sent.
Field Name: Q8ACMSGS

SENT - BLOCKS
The number of blocks sent.
Field Name: Q8ACBLKS

SENT - ROWS
The number of rows sent.
Field Name: Q8ACROWS

RECEIVED - BYTES
The number of bytes returned.
Field Name: Q8ACBYTR

RECEIVED - MESSAGES
The number of messages returned.
**Accelerator**

Field Name: Q8ACMSGR

**RECEIVED - BLOCKS**

The number of blocks returned.

Field Name: Q8ACBLKR

**RECEIVED - ROWS**

The number of rows returned.

Field Name: Q8ACROWR

**ELAPSED TIME - SVCS TCP/IP**

The accelerator services TCP/IP elapsed time measured in DB2. It starts when sending the requests to the accelerator and ends when receiving the results from the accelerator.

Field Name: Q8ACTELA

**ELAPSED TIME - ACCUM ACCEL**

The elapsed time spent in the accelerator when executing requests from the DB2 subsystem.

Field Name: Q8ACAELA

**CPU TIME - SVCS TCP/IP**

The accelerator services TCP/IP CPU time measured in DB2 for the amount of CPU consumed by the DDF service task to perform the SEND and RECEIVE to an accelerator service. It does not account for the TCP/IP address CPU to route the message on to the network and receive the reply into the DDF task.

Field Name: Q8ACTCPU

**CPU TIME - ACCUM ACCEL**

The CPU time spent in the accelerator when executing requests from the DB2 subsystem.

Field Name: Q8ACACPU

**WAIT TIME - ACCUM ACCEL**

The wait time spent in the accelerator when executing requests from the DB2 subsystem.

Field Name: Q8ACAWAT

**DB2 THREAD - CLASS 1 - ELAPSED**

Class 1 elapsed time of the thread. See ADRECETT.

Field Name: ADACCET1

**DB2 THREAD - CLASS 1 - CP CPU**

Class 1 CP CPU time of the thread. See ADCPUT.

Field Name: ADACCCP1

**DB2 THREAD - CLASS 1 - SE CPU**

Class 1 SE CPU time of the thread. See AWACC1Z.

Field Name: ADACCSE1
**DB2 THREAD - CLASS 2 - ELAPSED**

Class 2 elapsed time of the thread. See ADDB2ETT.

Field Name: ADACCET2

**DB2 THREAD - CLASS 2 - CP CPU**

Class 2 CP CPU time of the thread. See ADDBCPUT.

Field Name: ADACCCP2

**DB2 THREAD - CLASS 2 - SE CPU**

Class 2 SE CPU time of the thread. See AWACC2Z.

Field Name: ADACCSE2
Buffer Pool Activity

This topic shows detailed information about “Accounting - Buffer Pool Activity”.

This block is printed for each active buffer pool. When there is more than one active buffer pool, a block is printed for each aggregation showing total buffer pool activity (all buffer pools, all 4 KB buffer pools, all 32 KB buffer pools).

The following example applies to both, the report layout and the trace layout.

**Accounting - Buffer Pool Activity**

The field labels shown in the following sample layout of “Accounting - Buffer Pool Activity” are described in the following section.

<table>
<thead>
<tr>
<th>Report:</th>
<th>Trace:</th>
</tr>
</thead>
<tbody>
<tr>
<td>BP1</td>
<td>BPOOL ACTIVITY</td>
</tr>
<tr>
<td>---------</td>
<td>-----------------</td>
</tr>
<tr>
<td>BPOOL HIT RATIO (%)</td>
<td>9.14</td>
</tr>
<tr>
<td>GETPAGES</td>
<td>7.57</td>
</tr>
<tr>
<td>BUFFER UPDATES</td>
<td>0.00</td>
</tr>
<tr>
<td>SYNCHRONOUS WRITE</td>
<td>0.00</td>
</tr>
<tr>
<td>SYNCHRONOUS READ</td>
<td>2.71</td>
</tr>
<tr>
<td>SEQ. PREFETCH REQS</td>
<td>0.00</td>
</tr>
<tr>
<td>LIST PREFETCH REQS</td>
<td>0.00</td>
</tr>
<tr>
<td>DYN. PREFETCH REQS</td>
<td>0.00</td>
</tr>
<tr>
<td>PAGES READ ASYNCHR.</td>
<td>0.00</td>
</tr>
</tbody>
</table>

**BPOOL HIT RATIO (%)**

The percentage of Getpage operations that were satisfied by a page already in the buffer pool.

The value is calculated as the ratio of number of successful Getpage operations minus the number of pages read from DASD (both synchronously and using prefetch), to the number of successful Getpage operations, expressed as a percentage.

**Background and Tuning Information**

The highest possible hit ratio is 100%, that is, when every page requested is always in the buffer pool. If the requested page is not in the buffer pool, the hit ratio is 0% or less. If the hit ratio is negative, this means that prefetch brought pages into the buffer pool that are not subsequently referenced, either because the query stops before it reaches the end of the table space, or because the prefetched pages are stolen by DB2 for reuse before the query can access them. A low buffer pool hit ratio is not necessarily bad. The hit ratio is a relative value, based on the type of application. For example, an application that browses large data might have a buffer pool hit ratio of 0. Watch for those cases where the hit ratio drops significantly for the same application. Here are some suggestions to increase the buffer hit ratio:

- Run the REORG utility for indexes or table spaces associated with the virtual buffer pool.
- Reserve more pages for random I/O by setting the SEQUENTIAL STEAL THRESHOLD (VPSEQT) to a lower value.
- Increase the buffer pool as long as the cost of paging does not outweigh the benefit of I/O avoidance.
- Establish more separate buffer pools, perhaps to isolate different applications.
Buffer Pool Activity

The hit ratio measurement becomes less meaningful if the buffer pool is used by additional processes, such as utilities or work files.

Field Name: ABUFFRAT

GETPAGES

The number of Getpage requests. This counter is incremented by successful Getpage requests for queries processed in parallel for each thread and for all successful and unsuccessful Getpage requests for queries that are not processed in parallel.

Background and Tuning Information

Reducing the number of Getpages can improve DB2 performance by reducing the number of synchronous page reads. With fewer Getpages, the requested page is more likely to be returned from the buffer pool. CPU usage is also reduced.

Check the ratio of Getpages to SQL DML statements, as a rule of thumb, try and keep this ratio below six.

You might need to modify the database and query design, for example:

- Add indexes to tables to reduce the number of pages scanned.
- Reassess the number of tables used and denormalize them, if necessary.
  As an example, a large table with many columns can result in several pages being fetched to satisfy a simple query requesting just a few columns. Splitting such a table into several tables with fewer columns, tailored to queries, will result in fewer pages returned for each query.
- Use correlated rather than noncorrelated queries to force the use of an index.

Field Name: QBACGET

This is an exception field.

BUFFER UPDATES

The number of times a buffer update occurs. This is incremented every time a page is updated and is ready to be written to DASD. If the same page is updated twice, for example, the number is incremented by 2.

This number is kept for all types of pages including data pages and work-file pages.

Background and Tuning Information

A nonzero value indicates any of the following activities:

- SQL INSERT, UPDATE, or DELETE
- Merge scan join
- Internal sort activity on the work files

Check the access path to determine whether sort activity can be minimized or avoided.

Field Name: QBACSWS

This is an exception field.

SYNCHRONOUS WRITE

The number of immediate (synchronous) write I/O operations.

Background and Tuning Information
Buffer Pool Activity

Although an immediate write is rare, a small nonzero value is acceptable. A large value indicates that the system needs tuning.

Field Name: QBACIMW
This is an exception field.

SYNCHRONOUS READ

The number of synchronous read I/O operations. DB2 increments this counter for each media manager synchronous physical read. Asynchronous I/O requests are not counted.

Field Name: QBACRIO
This is an exception field.

SEQ. PREFETCH REQS

The number of SEQUENTIAL PREFETCH requests. This is incremented for each PREFETCH request. Each request can result in an I/O read. If it does, up to 32 pages can be read for SQL and up to 64 pages for utilities. For SQL, depending on the buffer pool size, a request does not result in an I/O if all the requested pages are already in the buffer pool.

DB2 can use sequential prefetch if the data is accessed in sequential order even though sequential prefetch was not requested at bind time. This is known as sequential detection and is not included in the sequential prefetch count. Sequential detection is included in dynamic prefetch requests field.

Background and Tuning Information

Table space scans and nonmatching index scans generally use sequential prefetch.

Field Name: QBACSEQ
This is an exception field.

LIST PREFETCH REQS

The number of LIST PREFETCH requests.

Special Considerations:
1. List prefetch allows DB2 to access data pages efficiently even if the needed data pages are not contiguous. It can be used with single index access and is always used with multiple index access.
2. List prefetch is always used to access data from the inner table during a hybrid join.
3. Data pages are read in quantities equal to the sequential prefetch quantity, which depends on the buffer pool size and is usually 32 pages.
4. During bind time DB2 does not use list prefetch if the estimated number of RIDs to be processed would take more than 50% of the RID pool. During execution time, list prefetch processing terminates if DB2 detects that more than 25% of the rows in the table need to be accessed. If list prefetch is terminated, it is indicated in IFCID 125.

Field Name: QBACLPF
This is an exception field.

DYN. PREFETCH REQS
**Buffer Pool Activity**

The number of (dynamic) PREFETCH requests. This is triggered by sequential detection. This includes prefetches for segmented table spaces.

**Background and Tuning Information**

Dynamic prefetch is typically used for a SELECT or UPDATE that is run repeatedly, accessing the index for each access.

If sequential prefetch, list prefetch, and dynamic prefetch reads have large values, check whether the access path can be improved.

**Field Name:** QBACDPF

This is an exception field.

**PAGES READ ASYNCHR.**

The number of asynchronous pages read by prefetch that the agent triggered.

**Background and Tuning Information**

This is used to determine the buffer pool hit ratio: (Getpage requests - Synchronous reads - Asynchronous pages read) / Getpage requests.

**Field Name:** QBACSIO

This is an exception field.
Data Capture

This topic shows detailed information about “Accounting - Data Capture”.

This block shows data for Data Capture activities.

For formatting reasons, OMEGAMON XE for DB2 PE shows different labels for report and trace. In the following example both layouts are shown, the report on the left, and the trace layout on the right.

Accounting - Data Capture

The field labels shown in the following sample layout of “Accounting - Data Capture” are described in the following section.

<table>
<thead>
<tr>
<th>Report:</th>
<th>Trace:</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATA CAPTURE</td>
<td>AVERAGE</td>
</tr>
<tr>
<td>IFI CALLS MADE</td>
<td>N/P</td>
</tr>
<tr>
<td>RECORDS CAPTURED</td>
<td>N/P</td>
</tr>
<tr>
<td>LOG RECORDS READ</td>
<td>N/P</td>
</tr>
<tr>
<td>ROWS RETURNED</td>
<td>N/P</td>
</tr>
<tr>
<td>RECORDS RETURNED</td>
<td>N/P</td>
</tr>
<tr>
<td>DATA DESC. RETURN</td>
<td>N/P</td>
</tr>
<tr>
<td>TABLES RETURNED</td>
<td>N/P</td>
</tr>
<tr>
<td>DESCRIBES</td>
<td>N/P</td>
</tr>
</tbody>
</table>

IFI CALLS MADE (IFI CALLS)

The total number of IFI calls. This field is only calculated if accounting class 5 is active.

Field Name: ADIFICAL

RECORDS CAPTURED (REC.CAPTURED)

The number of retrievable log records that were written for tables defined with DATA CAPTURE CHANGES. This number includes only those log records that can be retrieved by an IFI READS call for IFCID 185. Some records can be written but not retrieved, for example if monitor trace class 6 is not active.

Field Name: QIFAANRC

LOG RECORDS READ (LOG REC.READ)

The number of log reads performed for processing IFI READS requests for IFCID 185.

Field Name: QIFAANLR

ROWS RETURNED (ROWS RETURN)

The number of data rows returned in IFCID 185. Two rows are returned for each row altered by an SQL UPDATE statement.

Field Name: QIFAANDR

RECORDS RETURNED (RECORDS RET.)

The number of log records returned to the caller of the IFI READS call for IFCID 185.

Field Name: QIFAANRR

DATA DESC. RETURN (DATA DES.RET)
Data Capture

The number of data descriptions returned in IFCID 185. The data descriptions are mapped in IFCID 185.

Field Name: QIFAANDD

TABLES RETURNED (TABLES RET.)

The total number of tables returned to the caller of IFI READS call for IFCID 185.

Field Name: QIFAANTB

DESCRIBES

The number of data capture describes for processing READS requests for IFCID 185 data.

Field Name: QIFAANMB
Data Sharing Locking

This topic shows detailed information about “Accounting - Data Sharing Locking”.

This block shows the locking activity within a data sharing group.

For formatting reasons, OMEGAMON XE for DB2 PE shows different labels for report and trace. In the following example both layouts are shown, the report on the left, and the trace layout on the right.

Accounting - Data Sharing Locking

The field labels shown in the following sample layout of “Accounting - Data Sharing Locking” are described in the following section.

Report: Trace:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Average</th>
<th>Total</th>
<th>Field Name</th>
<th>Average</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATA SHARING AVERAGE TOTAL</td>
<td>--------</td>
<td>------</td>
<td>DATA SHARING TOTAL</td>
<td>--------</td>
<td>------</td>
</tr>
<tr>
<td>GLOBAL CONT RATE(%)</td>
<td>N/C</td>
<td>N/A</td>
<td>GLOB CONT(%)</td>
<td>11.35</td>
<td>------</td>
</tr>
<tr>
<td>FALSE CONT RATE(%)</td>
<td>N/C</td>
<td>N/A</td>
<td>FALS CONT(%)</td>
<td>0.00</td>
<td>------</td>
</tr>
<tr>
<td>P/L-LOCKS XES(%)</td>
<td>N/C</td>
<td>N/A</td>
<td>P/L-LOCKS(%)</td>
<td>89</td>
<td>------</td>
</tr>
<tr>
<td>LOCK REQ - PLOCKS</td>
<td>0.00</td>
<td>0</td>
<td>P-LOCK REQ</td>
<td>159</td>
<td>------</td>
</tr>
<tr>
<td>UNLOCK REQ - PLOCKS</td>
<td>0.00</td>
<td>0</td>
<td>P-UNLOCK REQ</td>
<td>45</td>
<td>------</td>
</tr>
<tr>
<td>CHANGE REQ - PLOCKS</td>
<td>0.00</td>
<td>0</td>
<td>P-CHANGE REQ</td>
<td>40</td>
<td>------</td>
</tr>
<tr>
<td>LOCK REQ - XES</td>
<td>0.00</td>
<td>0</td>
<td>LOCK - XES</td>
<td>286</td>
<td>------</td>
</tr>
<tr>
<td>UNLOCK REQ - XES</td>
<td>0.00</td>
<td>0</td>
<td>UNLOCK-XES</td>
<td>103</td>
<td>------</td>
</tr>
<tr>
<td>CHANGE REQ - XES</td>
<td>0.00</td>
<td>0</td>
<td>CHANGE-XES</td>
<td>47</td>
<td>------</td>
</tr>
<tr>
<td>SUSPENDS - IRLM</td>
<td>0.00</td>
<td>0</td>
<td>SUSP - IRLM</td>
<td>64</td>
<td>------</td>
</tr>
<tr>
<td>SUSPENDS - XES</td>
<td>0.00</td>
<td>0</td>
<td>SUSP - XES</td>
<td>0</td>
<td>------</td>
</tr>
<tr>
<td>CONVERSIONS - XES</td>
<td>0.00</td>
<td>0</td>
<td>CONV - XES</td>
<td>64</td>
<td>------</td>
</tr>
<tr>
<td>FALSE CONTENTIONS</td>
<td>0.00</td>
<td>0</td>
<td>FALSE CONT</td>
<td>0</td>
<td>------</td>
</tr>
<tr>
<td>INCOMPATIBLE LOCKS</td>
<td>0.00</td>
<td>0</td>
<td>INCOMP.LOCK</td>
<td>0</td>
<td>------</td>
</tr>
<tr>
<td>NOTIFY MSGS SENT</td>
<td>0.00</td>
<td>0</td>
<td>NOTIFY SENT</td>
<td>3</td>
<td>------</td>
</tr>
</tbody>
</table>

GLOBAL CONT RATE(%) (GLOB CONT(%))

The total number of suspends because of contention divided by the total number of synchronous requests that went to XES, and the lock requests that were converted from synchronous to asynchronous locks, and the locks because of child lock propagation.

Field Name: AGLOBRAT

FALSE CONT RATE(%) (FALS CONT(%) )

The total number of suspends because of false contention divided by the total number of synchronous requests that went to XES and the lock requests that were converted from synchronous to asynchronous locks.

A false contention is where two different locks on different resources hash to the same lock entry.

Field Name: AFLSERAT

P/L-LOCKS XES(%) (P/L-LOCKS (%) )

Shows the percentage of P/L-lock requests that were propagated to XES synchronously.

Background and Tuning Information

This number reflects the effects of explicit hierarchical locking and other locking optimizations. In an environment where all the workload is data sharing, a value of 94% means that 6% of all transaction locks were not propagated to XES due to Data-Sharing locking optimizations.
Data Sharing Locking

DB2 has optimizations to reduce the need to go beyond the local IRLM whenever possible:

- Explicit hierarchical locking allows IRLM to grant child locks locally when there is no inter-DB2 R/W interest on the parent.
- If there is a single DB2 with update interest, and multiple DB2s with read-only interest, DB2 propagates fewer locks than when all DB2s have update interest in the page set.
- All locks that go beyond the local IRLM are owned by the subsystem, not the individual work unit. This allows for further optimization. Only the most restrictive lock mode for an object on a given subsystem must be propagated to XES and the coupling facility. A new lock that is equally, or less, restrictive than the currently held lock is not propagated.

Field Name: ALLOCRAT

LOCK REQ - PLOCKS (P-LOCK REQ)

The number of lock requests for P-locks.

Field Name: QTGALPLK

UNLOCK REQ - PLOCKS (P-UNLOCK REQ)

The number of unlock requests for P-locks.

Field Name: QTGAUPLK

CHANGE REQ - PLOCKS (P-CHANGE REQ)

The number of change requests for P-locks.

Field Name: QTGACPLK

LOCK REQ - XES (LOCK - XES)

The number of P/L-lock requests propagated to z/OS XES synchronously.

This number is not incremented if the request is suspended before going to XES.

Field Name: QTGALSLM

UNLOCK REQ - XES (UNLOCK-XES)

The number of unlock requests propagated to z/OS XES.

Field Name: QTGAUSLM

CHANGE REQ - XES (CHANGE-XES)

The number of change requests propagated to z/OS XES.

Field Name: QTGACSLM

SUSPENDS - IRLM (SUSP - IRLM)

The number of suspensions due to IRLM global resource contention (IRLM lock states were in conflict).

Field Name: QTGAIGLO

SUSPENDS - XES (SUSP - XES)

The number of suspensions due to z/OS XES global resource contention (z/OS XES lock states were in conflict whereas IRLM lock states were not).

Field Name: QTGASGLO

SUSPENDS - CONV (SUSP - CONV)
Data Sharing Locking

The total number of sync-to-async heuristic conversions for LOCK requests in XES. This conversion is done when XES determines that it is more efficient to drive the request asynchronously to the coupling facility (CF).

Field Name: QTGAFLSE

FALSE CONTENTIONS

The total number of false contentions for LOCK and UNLOCK requests. A false contention occurs when different resource names hash to the same entry in the coupling facility (CF) lock table. The CF detects contention within the hash entry, and XES uses intersystem messaging to determine that no actual resource contention exists.

Field Name: QTGFCNT

INCOMPATIBLE LOCKS (INCOMPLOCK)

The number of global lock or change requests denied or suspended due to an incompatible retained lock.

Field Name: QTGADRTA

NOTIFY MSGS SENT (NOTIFY SENT)

The number of notify messages sent.

Field Name: QTGANTFY
Distributed Activity - Requester

This topic shows detailed information about “Accounting - Distributed Activity - Requester”.

This block shows the information provided for the requester of the distributed activity.

In the following example both layouts are shown, the report layout followed by the trace layout.

Accounting - Distributed Activity - Requester

The field labels shown in the following sample layout of “Accounting - Distributed Activity - Requester” are described in the following section.

Report:

--- DISTRIBUTED ACTIVITY
REQUESTER : *ROLSUM* #COMMIT(1) RECEIVED: 1111 MESSAGES SENT : 190.14 ROWS SENT : 0.00
PRODUCT ID : COMMON SERV #ROLLBK(1) RECEIVED: 43 MESSAGES RECEIVED: 190.14 BLOCKS SENT : 26.14
METHOD : DRDA PROTOCOL SQL RECEIVED : 136.68 BYTES SENT : 24298.55 #DDF ACCESSES: 22
CONV. INITIATED : 0.09 BYTES RECEIVED : 18850.82 #RLUP THREADS: 10
#COMMIT(2) RECEIVED: N/A TRANSACTIONS RECV. : N/A #PREPARE RECEIVED: N/A MSG.In BUFFER: N/A
#BACKOUT(2) RECEIVED: N/A #COMMIT(2) RES.SENT: N/A #LAST AGENT RECEV.: N/A #FORGET SENT : N/A
#COMMIT(2) PERFORM.: N/A #BACKOUT(2) RES.SENT: N/A #BACKOUT(2) PERFORM.: N/A
#BACKOUT(2) PERFORM.: N/A #MESSES IN BUFFER: N/A

Trace:

--- DISTRIBUTED ACTIVITY
REQUESTER : 9.164.162.248 ROLLBK(1) RECEIVED: 0.09 THREADS INDOUBT : 0
PRODUCT ID : COMMON SERV SQL RECEIVED : 136.68 ROWS SENT : 0
PRODUCT VERSION : MESSAGES SENT : 190.14 BLOCKS SENT : 0
METHOD : DRDA PROTOCOL MESSAGES RECEIVED: 190.14 CONVERSAT.INITIATED: 190.14
COMMIT(1) RECEIVED: 0 BYTES SENT : 24298.55 NBR RLUP THREADS : 0
COMMIT(2) RECEIVED : N/A COMMIT(2) RESP.SENT: N/A #PREPARE RECEIVED : N/A
BACKOUT(2) RECEIVED: N/A BACKOUT(2) RESP.SENT: N/A LAST AGENT RECEV.: N/A #FORGET SENT : N/A
BACKOUT(2) PERFORMED: N/A BACKOUT(2) PERFORMED: N/A MESSAGES IN BUFFER : N/A
COMMIT(2) PERFORMED: N/A TRANSACIONS RECV. : N/A

Report - REQUESTER

The name of the remote location with which this information is associated. If the local location is the requester, this field is a server location. If the local location is a server location, this field is the requester location. An allied thread is created at a DB2 requester, and a database access thread is created at a DB2 server. An accounting record is for either a requester or a server, but not for both.

This field is invalid in case of DB2 10 or later if summary rollup data is present. In Accounting this field is set to “ROLSUM”.

Field Name: QLACLOCN

This is an exception field.

Report - PRODUCT ID

The product ID and version of the remote location.

This field is invalid:

- In Accounting trace, it shows N/P.
- In Accounting report, it shows the last product ID being reduced, or hexadecimal 0 in case of DB2 10 or later and rollup summary data.
- In Accounting FILE and SAVE PROGRAM table, it shows blank.
Distributed Activity - Requester

- In case of DB2 10 or later if summary rollup data is present.

**Field Name:** QLACPRID

**Report - METHOD**

The method of access: DB2 private protocol, DRDA protocol, or both.

This field is invalid in case of DB2 10 or later if unique or summary rollup data is present. It can have the following value in:

- Accounting Trace and Report: N/P
- The Accounting FILE and SAVE PROGRAM table: blank

**Field Name:** ADPROTOD

**Report - CONV.INITIATED**

A count of conversations initiated by the requester.

This number is updated at the server location.

**Field Name:** QLACCNVR

**Report - #COMMIT(2) RECEIVED**

The number of commit requests received from the coordinator (two-phase commit operations only). This value is maintained at the participant, indicating that the participant was read only.

**Field Name:** QLACCRRC

**Report - #BCKOUT(2) RECEIVED**

The number of backout requests received from the coordinator (two-phase commit operations only). This value is maintained at the participant.

**Field Name:** QLACBKRC

**Report - #COMMIT(2) PERFORM.**

The number of commit operations performed with the remote location as coordinator (two-phase commit operations only). It is maintained at the participant.

**Field Name:** QLACCptr

**Report - #COMMIT(1) RECEIVED**

This field depends on the DB2 version that is installed:

- **DB2 10 or later:** The number of commit requests received from the requester (single-phase commit protocol) and committed requests received from the coordinator (two-phase commit protocol).
- **Prior to DB2 10:** The number of single-phase commit requests received from the requester location. This value is maintained at the server location.

**Field Name:** QLACCcomr

This is an exception field.

**Report - #ROLLBK(1) RECEIVED**

This field depends on the DB2 version that is installed:

- **DB2 10 or later:** The number of abort requests received from the requester (single-phase commit protocol) and backout requests received from the coordinator (two-phase commit protocol).
Distributed Activity - Requester

- **Prior to DB2 10:** The number of rollback requests received from the requester location (single-phase commit operations only). This value is maintained at the server location.

  **Field Name:** QLACABRR
  This is an *exception* field.

**Report - SQL RECEIVED**

The number of SQL statements received from the requester location.

  **Field Name:** QLACSQLR
  This is an *exception* field.

**Report - TRANSACTIONS RECV.**

The number of CREATE DATABASE ACCESS THREAD (DBAT) requests received by the server DBAT from the requester allied agent. This number is maintained by the server DBAT and is always 1.

  **Field Name:** QLACTRNR
  This is an *exception* field.

**Report - #COMMIT(2) RES.SENT**

The number of request commit responses sent to the coordinator (two-phase commit operations only). It is maintained at the participant.

  **Field Name:** QLACVYSE
  This is an *exception* field.

**Report - #BACKOUT(2)RES.SENT**

The number of backout responses sent to the coordinator (two-phase commit operations only). It is maintained at the participant and indicates that the participant rejected the PREPARE request.

  **Field Name:** QLACVNSE
  This is an *exception* field.

**Report - #BACKOUT(2)PERFORM.**

The number of rollback operations performed with the remote location as coordinator (two-phase commit operations only). It is maintained at the participant.

  **Field Name:** QLACRBTR
  This is an *exception* field.

**Report - MESSAGES SENT**

The number of messages sent to the location. It is maintained at the location where the messages originated.

  **Field Name:** QLACMSGS
  This is an *exception* field.

**Report - MESSAGES RECEIVED**

The number of messages received from the location. This value is maintained at the location where the messages were received.
Distributed Activity - Requester

More messages might be sent from the server location than are received by the requester because of the way in which distributed SQL statements are processed internally.

Field Name: QLACMSGR
This is an exception field.

Report - BYTES SENT
The number of bytes the server location sent to the requester location. This value is maintained at the server location.

More bytes of data might be sent from the server location than are received by the requester due to the way in which distributed SQL statements are processed internally.

Field Name: QLACBYTS
This is an exception field.

Report - BYTES RECEIVED
The number of bytes the server location received from the requester location.

More bytes of data might be sent from the server location than are received by the requester, because of the way in which distributed SQL statements are processed internally.

Field Name: QLACBYTR
This is an exception field.

Report - #THREADS INDOUBT
The number of threads that went indoubt with the remote location as coordinator (two-phase commit operations only). It is maintained at the participant and indicates that the communication with the coordinator was lost.

Field Name: QLACINDT
This is an exception field.

Report - #PREPARE RECEIVED
The number of PREPARE requests received from the coordinator (two-phase commit operations only). It is maintained at the participant.

Field Name: QLACPRRC
This is an exception field.

Report - #LAST AGENT RECV.
The number of last agent requests received from the initiator (two-phase commit operations only). It is maintained at the participant.

Field Name: QLACLARC
This is an exception field.

Report - ROWS SENT
The number of rows sent from the server location to the requester location.
The value includes SQLDA and is maintained at the server location.

Field Name: QLACROWS
Distributed Activity - Requester

This is an exception field.

Report - BLOCKS SENT
The number of blocks transmitted using block fetch. This value is maintained at the server location.

Field Name: QLACBTBF
This is an exception field.

Report - #DDF ACCESSES
The number of occurrences of the remote location and method pair.

Field Name: ASDDF
This is an exception field.

Report - #RLUP THREADS
The number of threads to roll data into this QLAC data section. Non-rollup QLACs have a value of 1 and rollup QLACs have a value of 1 or more.

Field Name: QLACRLNU

Report - MSG.IN BUFFER
The number of rows transmitted or received in DB2 message buffers using block fetch. This includes both requester and server activity.

Field Name: QLACBROW
This is an exception field.

Report - #FORGET SENT
The number of forget responses sent to the coordinator (two-phase commit operations only). It is maintained at the participant.

Field Name: QLACRRSE
This is an exception field.

Trace - REQUESTER
The name of the remote location with which this information is associated. If the local location is the requester, this field is a server location. If the local location is a server location, this field is the requester location. An allied thread is created at a DB2 requester, and a database access thread is created at a DB2 server. An accounting record is for either a requester or a server, but not for both.

This field is invalid in case of DB2 10 or later if summary rollup data is present. In Accounting this field is set to "ROLSUM".

Field Name: QLACLOCN
This is an exception field.

Trace - PRODUCT ID (PRODUCT VERSION)
The product ID and version of the remote location.

This field is invalid:

- In Accounting trace, it shows N/P.
- In Accounting report, it shows the last product ID being reduced, or hexadecimal 0 in case of DB2 10 or later and rollup summary data.
- In Accounting FILE and SAVE PROGRAM table, it shows blank.
Distributed Activity - Requester

- In case of DB2 10 or later if summary rollup data is present.
  
  **Field Name**: QLACPRID

**Trace - METHOD**

The method of access: DB2 private protocol, DRDA protocol, or both.

This field is invalid in case of DB2 10 or later if unique or summary rollup data is present. It can have the following value in:
- Accounting Trace and Report: N/P
- The Accounting FILE and SAVE PROGRAM table: blank

**Field Name**: ADPROT OC

**Trace - COMMITS(1) RECEIVED**

This field depends on the DB2 version that is installed:
- **DB2 10 or later**: The number of commit requests received from the requester (single-phase commit protocol) and committed requests received from the coordinator (two-phase commit protocol).
- **Prior to DB2 10**: The number of single-phase commit requests received from the requester location. This value is maintained at the server location.

**Field Name**: QLACCOMR

This is an exception field.

**Trace - COMMIT(2) RECEIVED**

The number of commit requests received from the coordinator (two-phase commit operations only). This value is maintained at the participant, indicating that the participant was read only.

**Field Name**: QLACCRRC

This is an exception field.

**Trace - BACKOUT(2) RECEIVED**

The number of backout requests received from the coordinator (two-phase commit operations only). This value is maintained at the participant.

**Field Name**: QLACBKRC

This is an exception field.

**Trace - COMMIT(2) PERFORMED**

The number of commit operations performed with the remote location as coordinator (two-phase commit operations only). It is maintained at the participant.

**Field Name**: QLACC PTR

This is an exception field.

**Trace - TRANSACTIONS RECV.**

The number of CREATE DATABASE ACCESS THREAD (DBAT) requests received by the server DBAT from the requester allied agent. This number is maintained by the server DBAT and is always 1.

**Field Name**: QLACTRNR

This is an exception field.
Trace - ROLLBCK(1) RECEIVED

This field depends on the DB2 version that is installed:

- **DB2 10 or later:** The number of abort requests received from the requester (single-phase commit protocol) and backout requests received from the coordinator (two-phase commit protocol).
- **Prior to DB2 10:** The number of rollback requests received from the requester location (single-phase commit operations only). This value is maintained at the server location.

**Field Name:** QLACABRR

This is an exception field.

Trace - SQL RECEIVED

The number of SQL statements received from the requester location.

**Field Name:** QLACSQLR

This is an exception field.

Trace - MESSAGES SENT

The number of messages sent to the location. It is maintained at the location where the messages originated.

**Field Name:** QLACMSGS

This is an exception field.

Trace - MESSAGES RECEIVED

The number of messages received from the location. This value is maintained at the location where the messages were received.

More messages might be sent from the server location than are received by the requester because of the way in which distributed SQL statements are processed internally.

**Field Name:** QLACMSGR

This is an exception field.

Trace - BYTES SENT

The number of bytes the server location sent to the requester location. This value is maintained at the server location.

More bytes of data might be sent from the server location than are received by the requester due to the way in which distributed SQL statements are processed internally.

**Field Name:** QLACBYTS

This is an exception field.

Trace - BYTES RECEIVED

The number of bytes the server location received from the requester location.

More bytes of data might be sent from the server location than are received by the requester, because of the way in which distributed SQL statements are processed internally.

**Field Name:** QLACBYTR
Distributed Activity - Requester

This is an exception field.

Trace - COMMIT(2) RESP.SENT
The number of request commit responses sent to the coordinator (two-phase commit operations only). It is maintained at the participant.

Field Name: QLACVYSE
This is an exception field.

Trace - BACKOUT(2)RESP.SENT
The number of backout responses sent to the coordinator (two-phase commit operations only). It is maintained at the participant and indicates that the participant rejected the PREPARE request.

Field Name: QLACVNSE
This is an exception field.

Trace - BACKOUT(2)PERFORMED
The number of rollback operations performed with the remote location as coordinator (two-phase commit operations only). It is maintained at the participant.

Field Name: QLACRBTR
This is an exception field.

Trace - THREADS INDOUBT
The number of threads that went indoubt with the remote location as coordinator (two-phase commit operations only). It is maintained at the participant and indicates that the communication with the coordinator was lost.

Field Name: QLACINDT
This is an exception field.

Trace - ROWS SENT
The number of rows sent from the server location to the requester location. The value includes SQLDA and is maintained at the server location.

Field Name: QLACROWS
This is an exception field.

Trace - BLOCKS SENT
The number of blocks transmitted using block fetch. This value is maintained at the server location.

Field Name: QLACBTBF
This is an exception field.

Trace - CONV.INITIATED (CONVERSAT.INITIATED)
A count of conversations initiated by the requester. This number is updated at the server location.

Field Name: QLACCNVR
This is an exception field.

Trace - NBR RLUP THREADS
The number of threads to roll data into this QLAC data section. Non-rollup QLACs have a value of 1 and rollup QLACs have a value of 1 or more.

**Field Name:** QLACRLNU

**Trace - PREPARE RECEIVED**

The number of PREPARE requests received from the coordinator (two-phase commit operations only). It is maintained at the participant.

**Field Name:** QLACPRRC

**Trace - LAST AGENT RECV.**

The number of last agent requests received from the initiator (two-phase commit operations only). It is maintained at the participant.

**Field Name:** QLACLARC

**Trace - MESSAGES IN BUFFER**

The number of rows transmitted or received in DB2 message buffers using block fetch. This includes both requester and server activity.

**Field Name:** QLACBROW

This is an exception field.

**Trace - FORGET SENT**

The number of forget responses sent to the coordinator (two-phase commit operations only). It is maintained at the participant.

**Field Name:** QLACRRSE

This is an exception field.
Distributed Activity - Server

This topic shows detailed information about "Accounting - Distributed Activity - Server".

This block shows the information provided for the server of the distributed activity.

In the following example both layouts are shown, the report layout followed by the trace layout.

Accounting - Distributed Activity - Server

The field labels shown in the following sample layout of "Accounting - Distributed Activity - Server" are described in the following section.

Report:

```
---- DISTRIBUTED ACTIVITY
PRODUCT ID : DB2
METHOD : DB2 PRIV
REQUESTER ELAP. TIME: 10.776739
SERVER ELAPSED TIME: 2.952933
SERVER CPU TIME: 0.014974
DAT WATING TIME: 0.010000

#COMMIT(1)SENT: 0 #ROLLB(1)SENT: 0 MESSAGES SENT: 3.00
CONVERSATIONS INITIATED: 1.00
CONVERSATIONS QUEUED: 0
MESSAGES RECEIVED: 3.00

SQL SENT: 2.00 BYTES SENT: 1314.00
ROWS RECEIVED: 20.00 BLOCKS RECEIVED: 1.00
BYTES RECEIVED: 2076.00

#COMMIT(2) SENT: N/A
SUCCESSFULLY ALLOC.COMV: N/A
TRANSACT.SENT: N/A
STMT BOUND AT SER: N/A

#COMMIT(2) RESP.RECVR: N/A

--- DISTRIBUTED ACTIVITY
SERVER : PM02DE21
CONVERSATION TERMINATED: 0.00
NBR RLUP THREADS: 0

PRODUCT ID : DB2
METHOD : DB2 PRIV
REQUESTER ELAP. TIME: 10.776739
SERVER ELAPSED TIME: 10.776739
SERVER CPU TIME: 0.014974
DAT WATING TIME: 0.010000

CONVERSATIONS INITIATED: 0
CONVERSATIONS QUEUED: 0

#COMMIT(2) SENT: N/A
SUCCESSFULLY ALLOC.COMV: N/A
MSG.IN BUFFER: N/A

#COMMIT(2) RESP.RECVR: N/A

```

Trace:

```
---- DISTRIBUTED ACTIVITY
PRODUCT ID : DB2
METHOD : DB2 PRIV
REQUESTER ELAP. TIME: 10.776739
SERVER ELAPSED TIME: 2.952933
SERVER CPU TIME: 0.014974
DAT WATING TIME: 0.010000

#COMMIT(1)SENT: 0 #ROLLB(1)SENT: 0 MESSAGES SENT: 3.00
CONVERSATIONS INITIATED: 1.00
CONVERSATIONS QUEUED: 0
MESSAGES RECEIVED: 3.00

SQL SENT: 2.00 BYTES SENT: 1314.00
ROWS RECEIVED: 20.00 BLOCKS RECEIVED: 1.00
BYTES RECEIVED: 2076.00

#COMMIT(2) SENT: N/A
SUCCESSFULLY ALLOC.COMV: N/A
TRANSACT.SENT: N/A
STMT BOUND AT SER: N/A

#COMMIT(2) RESP.RECVR: N/A

```

Report - SERVER

The name of the remote location with which this information is associated.
If the local location is the requester, this field is a server location. If the local location is a server location, this field is the requester location. An allied thread is created at a DB2 requester, and a database access thread is created at a DB2 server. An accounting record is for either a requester or a server, but not for both.

This field is invalid in case of DB2 10 or later if summary rollup data is present. In Accounting this field is set to "ROLSUM".

Field Name: QLACLOCN

This is an exception field.

Report - PRODUCT ID

The product ID and version of the remote location.
This field is invalid:
- In Accounting trace, it shows N/P.
- In Accounting report, it shows the last product ID being reduced, or hexadecimal 0 in case of DB2 10 or later and rollup summary data.
- In Accounting FILE and SAVE PROGRAM table, it shows blank.
- In case of DB2 10 or later if summary rollup data is present.

Field Name: QLACPRID

Report - METHOD
The method of access: DB2 private protocol, DRDA protocol, or both.
This field is invalid in case of DB2 10 or later if unique or summary rollup data is present. It can have the following value in:
- Accounting Trace and Report: N/P
- The Accounting FILE and SAVE PROGRAM table: blank

Field Name: ADPROTOC

Report - REQUESTER ELAP.TIME
The elapsed time at the requester. It includes the total of DB2 and network time.

Field Name: ADDSELRQ

Report - SERVER ELAPSED TIME
The elapsed database access agent time at the server location. This value is updated at the requester location.

Special Considerations:
- This value is reported only for DB2 private protocol. If only DRDA protocol, N/C is shown.
- If both DB2 private protocol and DRDA protocol are used, then only the elapsed time associated with the DB2 private protocol is reported, and this can be misleading.
- This value is calculated by accumulating the difference between the store clock values obtained after receiving a request message and before sending the associated reply message.
- When block fetch is used, this time can be longer than the time for ADDSELRQ (ELAPSED REQ).
- Compare this value with the accounting class 2 time (allied agent time in DB2) to see if the distributed-allied thread using the database access agent spends too much time in remote processing.

Field Name: ADDSELSR
This is an exception field.

Report - SERVER CPU TIME
The database access agent CPU time spent at the server location. This value is updated at the requester location, is intended for problem determination only, and should not be used for charge back.

Special Considerations:
1. This value is reported only for DB2 private protocol. If only DRDA protocol is used, N/C is shown.
2. If both DB2 private protocol and DRDA protocol are used, then only the CPU time associated with the DB2 private protocol is reported, and this can be misleading.

3. This value is calculated by accumulating the amount of CPU time spent by the database access thread at the DB2 server location each time a request message is processed.

4. Certain programming techniques can cause this value to not be received at the requester location (and therefore not included in this field), even though the CPU time was spent at the server location and was properly measured and sent to the requester location.

Field Name: ADDSSRSR
This is an exception field.

Report - DBAT WAITING TIME
Total elapsed time spent waiting for an available database access agent slot.
Field Name: QLACMDW

Report - #DDF ACCESSES
The number of occurrences of the remote location and method pair.
Field Name: ASDDF

Report - #COMMIT(2) SENT
The number of commit requests sent to the participant (two-phase commit operations only). This value is maintained at the participant, indicating that the participant was read only.
Field Name: QLACC

Report - CONVERSATIONS INITIATED
The number of conversations (both successful and unsuccessful) initiated by the requester location to be executed at the server location. This number is maintained at the requester.
Field Name: QLACCN

Report - #CONVERSATIONS QUEUED
A number of conversation requests queued by DDF that are waiting for allocation. This value is maintained at the requester location.
If the value is a large number, you might want to increase the limit for the number of conversations.
Field Name: QLACCNVQ
This is an exception field.

Report - CONVERSATION TERMINATED
The number of terminated conversations in the server block (DB2 private protocol only). It is maintained at the requester location.
This number can be different from the number of successful conversation allocations, because some conversations might not have been terminated when the accounting record was written.
Field Name: QLACCNT
Distributed Activity - Server

This is an exception field.

Report - #RLUP THREADS
The number of threads to roll data into this QLAC data section. Non-rollup QLACs have a value of 1 and rollup QLACs have a value of 1 or more.

Field Name: QLACRLNU

Report - #BACKOUT(2) SENT
The number of backout requests sent to the participant (two-phase commit operations only).

Field Name: QLACBKSE

Report - SUCCESSFULLY ALLOC.CONV
The number of successful conversation allocations made to the server (DB2 private protocol only). This value is maintained at the requester location.

All allocation attempts, whether successful or not, are counted in QLACCNVS. The difference between QLACCNVS and this field helps to identify session resource constraint problems. Counting the number of unsuccessful conversations is useful for session tuning.

Field Name: QLACCNV

Report - MAX OPEN CONVERSATIONS
The maximum number of conversations open at any time (QLACCNV - QLACCNVT). QLACCIEL is updated only when (QLACCNV - QLACCNVT) is greater than the current value of QLACCIEL. QLACFLG1 and QLACFLG2 indicate whether the conversations use DB2 private protocol, DRDA protocol, or both. This value is maintained at the requester location.

Field Name: QLACCIEL

Report - #CONT->LIM.BL.FTCH SWCH
The number of times continuous block mode switched to limited block mode (DB2 private protocol only).

Field Name: QLACCBLB

Report - #COMMIT(2) RESP.RECV.
The number of request commit responses received from the participant (two-phase commit operations only). It is maintained at the coordinator.

Field Name: QLACVYRC

Report - #COMMIT(1)SENT
This field depends on the DB2 version that is installed:

- **DB2 10 or later**: The number of commit requests sent to the server (single-phase commit protocol) and committed requests sent to the participant (two-phase commit protocol).

- **Prior to DB2 10**: The number of single-phase commit requests sent to the server location. This value is maintained at the requester location.

Field Name: QLACCOMS

This is an exception field.

Report - #ROLLB(1)SENT
Distributed Activity - Server

This field depends on the DB2 version that is installed:

- **DB2 10 or later:** The number of abort requests sent to the server (single-phase commit protocol) and backout requests sent to the participant (two-phase commit protocol).
- **Prior to DB2 10:** The number of rollback requests sent to the server location (single-phase commit operations only). This value is maintained at the requester location.

Field Name: QLACABRS

This is an *exception* field.

Report - SQL SENT

The number of SQL statements sent to the server location. This value is maintained at the requesting location.

Field Name: QLACSQLS

This is an *exception* field.

Report - ROWS RECEIVED

The number of rows of data retrieved from the server location. This value is maintained at the requester location.

*Special Considerations:*

1. The number of rows received from the server location does not include either the SQLDA or SQLCA.
2. Block fetch can significantly affect the number of rows sent across the network. When used with non-UPDATE cursors, block fetch puts as many rows as possible into the message buffer, and transmits the buffer across the network without requiring a VTAM message. Consequently, more rows of data might be sent from the server location than are received by the reporting (requester) location. This is especially true when DB2 private protocol is used because multiple blocks can be transmitted from the server with no intervening messages sent by the requester.

Field Name: QLACROWR

This is an *exception* field.

Report - #BKOUT(2) R.R

The number of backout responses received from the participant (two-phase commit operations only). It is maintained at the coordinator and indicates that the participant rejected the PREPARE request.

Field Name: QLACVNRC

This is an *exception* field.

Report - TRANSACT.SENT

The number of CREATE DATABASE ACCESS THREAD (DBAT) requests the requester allied agent sent to the server location. This number is maintained by the requester allied agent.

In some cases, for example when a new user signs on or a resignon occurs, the value of this field can be zero. This indicates that the existing DBAT at the server was reused by this user.

Field Name: QLACTRNS
Distributed Activity - Server

This is an exception field.

Report - MSG.IN BUFFER
The number of rows transmitted or received in DB2 message buffers using block fetch. This includes both requester and server activity.

Field Name: QLACBROW
This is an exception field.

Report - #PREPARE SENT
The number of PREPARE requests sent to the participant (two-phase commit operations only). It is maintained at the coordinator.

Field Name: QLACPRSE
This is an exception field.

Report - MESSAGES SENT
The number of messages sent to the location. It is maintained at the location where the messages originated.

Field Name: QLACMSGS
This is an exception field.

Report - MESSAGES RECEIVED
The number of messages received from the location. This value is maintained at the location where the messages were received.

More messages might be sent from the server location than are received by the requester because of the way in which distributed SQL statements are processed internally.

Field Name: QLACMSGR
This is an exception field.

Report - BYTES SENT
The number of bytes the server location sent to the requester location. This value is maintained at the server location.

More bytes of data might be sent from the server location than are received by the requester due to the way in which distributed SQL statements are processed internally.

Field Name: QLACBYTS
This is an exception field.

Report - BYTES RECEIVED
The number of bytes the server location received from the requester location.

More bytes of data might be sent from the server location than are received by the requester, because of the way in which distributed SQL statements are processed internally.

Field Name: QLACBYTR
This is an exception field.

Report - BLOCKS RECEIVED
Distributed Activity - Server

The number of blocks received using block fetch. This value is maintained at the requester location.

Field Name: QLACBRBF

This is an exception field.

Report - #LASTAGN.SENT (LAST AGN.SENT)

The number of last agent requests sent to the coordinator (two-phase commit operations only).

A last agent request reduces the number of messages that must be sent for the commit. If DB2 is the requester, this number is incremented when a conversation is deallocated and this conversation was not used since the last commit. If this number is large, and your application design permits it, you can save another message by issuing a release before the commit (only for a DB2 requester).

Field Name: QLACLASE

Report - STMT BOUND AT SER

The number of static SQL statements that were bound for remote access (DB2 private protocol only). This value is maintained at the requester location.

Field Name: QLACRBND

Report - #FORGET RECEIVED

The number of forget responses received from the participant (two-phase commit operations only). It is maintained at the coordinator.

Field Name: QLACRRRC

Trace - SERVER

The name of the remote location with which this information is associated. If the local location is the requester, this field is a server location. If the local location is a server location, this field is the requester location. An allied thread is created at a DB2 requester, and a database access thread is created at a DB2 server. An accounting record is for either a requester or a server, but not for both.

This field is invalid in case of DB2 10 or later if summary rollup data is present. In Accounting this field is set to *ROLSUM*.

Field Name: QLACLOCN

This is an exception field.

Trace - PRODUCT ID (PRODUCT VERSION)

The product ID and version of the remote location.

This field is invalid:

- In Accounting trace, it shows N/P.
- In Accounting report, it shows the last product ID being reduced, or hexadecimal 0 in case of DB2 10 or later and rollup summary data.
- In Accounting FILE and SAVE PROGRAM table, it shows blank.
- In case of DB2 10 or later if summary rollup data is present.

Field Name: QLACPRID

Trace - METHOD
The method of access: DB2 private protocol, DRDA protocol, or both. This field is invalid in case of DB2 10 or later if unique or summary rollup data is present. It can have the following value in:

- Accounting Trace and Report: N/P
- The Accounting FILE and SAVE PROGRAM table: blank

**Field Name:** ADPROT0C

**Trace - REQUESTER ELAP.TIME**

The elapsed time at the requester. It includes the total of DB2 and network time.

**Field Name:** ADDSELRQ

**Trace - SERVER ELAPSED TIME**

The elapsed database access agent time at the server location. This value is updated at the requester location.

**Special Considerations:**

- This value is reported only for DB2 private protocol. If only DRDA protocol, N/C is shown.
- If both DB2 private protocol and DRDA protocol are used, then only the elapsed time associated with the DB2 private protocol is reported, and this can be misleading.
- This value is calculated by accumulating the difference between the store clock values obtained after receiving a request message and before sending the associated reply message.
- When block fetch is used, this time can be longer than the time for ADDSELRQ (ELAPSED REQ).
- Compare this value with the accounting class 2 time (allied agent time in DB2) to see if the distributed-allied thread using the database access agent spends too much time in remote processing.

**Field Name:** ADDSELSR

This is an exception field.

**Trace - SERVER CPU TIME**

The database access agent CPU time spent at the server location. This value is updated at the requester location, is intended for problem determination only, and should not be used for charge back.

**Special Considerations:**

1. This value is reported only for DB2 private protocol. If only DRDA protocol is used, N/C is shown.
2. If both DB2 private protocol and DRDA protocol are used, then only the CPU time associated with the DB2 private protocol is reported, and this can be misleading.
3. This value is calculated by accumulating the amount of CPU time spent by the database access thread at the DB2 server location each time a request message is processed.
4. Certain programming techniques can cause this value to not be received at the requester location (and therefore not included in this field), even though the CPU time was spent at the server location and was properly measured and sent to the requester location.
Distributed Activity - Server

Field Name: ADDSSRSR
This is an exception field.

Trace - DBAT WAITING TIME
Total elapsed time spent waiting for an available database access agent slot.
Field Name: QLACMDWT

Trace - CONVERSATIONS INITIATED
The number of conversations (both successful and unsuccessful) initiated by the requester location to be executed at the server location. This number is maintained at the requester.
Field Name: QLACCNVS

Trace - CONVERSATIONS QUEUED
A number of conversation requests queued by DDF that are waiting for allocation. This value is maintained at the requester location.
If the value is a large number, you might want to increase the limit for the number of conversations.
Field Name: QLACCNVQ
This is an exception field.

Trace - #COMMIT(2) SENT
The number of commit requests sent to the participant (two-phase commit operations only). This value is maintained at the participant, indicating that the participant was read only.
Field Name: QLACCRSE
This is an exception field.

Trace - #BACKOUT(2) SENT
The number of backout requests sent to the participant (two-phase commit operations only).
Field Name: QLACBKSE
This is an exception field.

Trace - CONVERSATION TERMINATED
The number of terminated conversations in the server block (DB2 private protocol only). It is maintained at the requester location.
This number can be different from the number of successful conversation allocations, because some conversations might not have been terminated when the accounting record was written.
Field Name: QLACCNVT
This is an exception field.

Trace - COMMIT(1)SENT
This field depends on the DB2 version that is installed:
- **DB2 10 or later:** The number of commit requests sent to the server (single-phase commit protocol) and committed requests sent to the participant (two-phase commit protocol).
Prior to DB2 10:
The number of single-phase commit requests sent to the server location. This value is maintained at the requester location.

Field Name: QLACCOMS
This is an exception field.

Trace - ROLLB(1)SENT
This field depends on the DB2 version that is installed:

- **DB2 10 or later:** The number of abort requests sent to the server (single-phase commit protocol) and backout requests sent to the participant (two-phase commit protocol).
- **Prior to DB2 10:** The number of rollback requests sent to the server location (single-phase commit operations only). This value is maintained at the requester location.

Field Name: QLACABRS
This is an exception field.

Trace - SQL SENT
The number of SQL statements sent to the server location. This value is maintained at the requesting location.

Field Name: QLACSQLS
This is an exception field.

Trace - ROWS RECEIVED
The number of rows of data retrieved from the server location. This value is maintained at the requester location.

Special Considerations:
1. The number of rows received from the server location does not include either the SQLDA or SQLCA.
2. Block fetch can significantly affect the number of rows sent across the network. When used with non-UPDATE cursors, block fetch puts as many rows as possible into the message buffer, and transmits the buffer across the network without requiring a VTAM message. Consequently, more rows of data might be sent from the server location than are received by the reporting (requester) location. This is especially true when DB2 private protocol is used because multiple blocks can be transmitted from the server with no intervening messages sent by the requester.

Field Name: QLACROWR
This is an exception field.

Trace - SUCCESSFULLY ALLOC.CONV
The number of successful conversation allocations made to the server (DB2 private protocol only). This value is maintained at the requester location.

All allocation attempts, whether successful or not, are counted in QLACCNVS. The difference between QLACCNVS and this field helps to identify session resource constraint problems. Counting the number of unsuccessful conversations is useful for session tuning.

Field Name: QLACCNVA

Trace - MAX OPEN CONVERSATIONS
Distributed Activity - Server

The maximum number of conversations open at any time (QLACCNVA - QLACCNVT). QLACCIEL is updated only when (QLACCNVA - QLACCNVT) is greater than the current value of QLACCIEL. QLACFLG1 and QLACFLG2 indicate whether the conversations use DB2 private protocol, DRDA protocol, or both. This value is maintained at the requester location.

Field Name: QLACCIEL

Trace - CONT->LIM.BL.FTCH SWTCH

The number of times continuous block mode switched to limited block mode (DB2 private protocol only).

Field Name: QLACCBLB

Trace - #COMMIT(2) RESP.RECV. (COMMIT(2) RESP.RECEIVED)

The number of request commit responses received from the participant (two-phase commit operations only). It is maintained at the coordinator.

Field Name: QLACVYRC

Trace - #BKOUT(2) R.R (BKOUT(2) R.R)

The number of backout responses received from the participant (two-phase commit operations only). It is maintained at the coordinator and indicates that the participant rejected the PREPARE request.

Field Name: QLACVNRC

Trace - TRANSACT.SENT

The number of CREATE DATABASE ACCESS THREAD (DBAT) requests the requester allied agent sent to the server location. This number is maintained by the requester allied agent.

In some cases, for example when a new user signs on or a resignon occurs, the value of this field can be zero. This indicates that the existing DBAT at the server was reused by this user.

Field Name: QLACTRNS

Trace - NBR RLUP THREADS

The number of threads to roll data into this QLAC data section. Non-rollup QLACs have a value of 1 and rollup QLACs have a value of 1 or more.

Field Name: QLACRLNU

Trace - MESSAGES SENT

The number of messages sent to the location. It is maintained at the location where the messages originated.

Field Name: QLACMSGS

Trace - MESSAGES RECEIVED

The number of messages received from the location. This value is maintained at the location where the messages were received.

More messages might be sent from the server location than are received by the requester because of the way in which distributed SQL statements are processed internally.

Field Name: QLACMSGR
Distributed Activity - Server

Trace - BYTES SENT
The number of bytes the server location sent to the requester location. This value is maintained at the server location.

More bytes of data might be sent from the server location than are received by the requester due to the way in which distributed SQL statements are processed internally.

Field Name: QLACBYTS

Trace - BYTES RECEIVED
The number of bytes the server location received from the requester location.

More bytes of data might be sent from the server location than are received by the requester, because of the way in which distributed SQL statements are processed internally.

Field Name: QLACBYTR

Trace - BLOCKS RECEIVED
The number of blocks received using block fetch. This value is maintained at the requester location.

Field Name: QLACBRBF

Trace - MSG.IN BUFFER
The number of rows transmitted or received in DB2 message buffers using block fetch. This includes both requester and server activity.

Field Name: QLACBROW

This is an exception field.

Trace - PREPARE SENT
The number of PREPARE requests sent to the participant (two-phase commit operations only). It is maintained at the coordinator.

Field Name: QLACPRSE

This is an exception field.

Trace - LAST AGN.SENT
The number of last agent requests sent to the coordinator (two-phase commit operations only).

A last agent request reduces the number of messages that must be sent for the commit. If DB2 is the requester, this number is incremented when a conversation is deallocated and this conversation was not used since the last commit. If this number is large, and your application design permits it, you can save another message by issuing a release before the commit (only for a DB2 requester).

Field Name: QLACLASE

Trace - STMT BOUND AT SER
The number of static SQL statements that were bound for remote access (DB2 private protocol only). This value is maintained at the requester location.

Field Name: QLACRBND
Distributed Activity - Server

Trace - FORGET RECEIVED

The number of forget responses received from the participant (two-phase commit operations only). It is maintained at the coordinator.

Field Name: QLACRRRC
Drain and Claim

This topic shows detailed information about “Accounting - Drain and Claim”.

This block contains information about requesting a drain or a claim.

For formatting reasons, OMEGAMON XE for DB2 PE shows different labels for report and trace. The following example shows both layouts, the report on the left, and the trace layout on the right.

**Accounting - Drain and Claim**

The field labels shown in the following sample layout of “Accounting - Drain and Claim” are described in the following section.

<table>
<thead>
<tr>
<th>Report:</th>
<th>Trace:</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DRAIN/CLAIM</strong></td>
<td><strong>AVERAGE TOTAL</strong></td>
</tr>
<tr>
<td>DRAIN REQUESTS</td>
<td>0.00 0</td>
</tr>
<tr>
<td>DRAIN FAILED</td>
<td>0.00 0</td>
</tr>
<tr>
<td>CLAIM REQUESTS</td>
<td>15.00 15</td>
</tr>
<tr>
<td>CLAIM FAILED</td>
<td>0.00 0</td>
</tr>
</tbody>
</table>

**DRAIN REQUESTS (DRAIN REQST)**

The number of drain requests.

**Field Name:** QTXADRNO

**DRAIN FAILED**

The number of unsuccessful drain requests.

**Field Name:** QTXADRUN

**CLAIM REQUESTS (CLAIM REQST)**

The number of claim requests.

**Field Name:** QTXACLNO

**CLAIM FAILED**

The number of unsuccessful claim requests.

**Field Name:** QTXACLUN
Dynamic SQL Statement

This topic shows detailed information about “Accounting - Dynamic SQL Statement”.

This block provides information about the dynamic SQL statement.

The following example shows both layouts, the report on the left, and the trace layout on the right.

Accounting - Dynamic SQL Statement

The field labels shown in the following sample layout of “Accounting - Dynamic SQL Statement” are described in the following section.

**Report:**

<table>
<thead>
<tr>
<th>Field</th>
<th>Average</th>
<th>Total</th>
<th>Field</th>
<th>Average</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>DYNAMIC SQL STMT AVERAGE</td>
<td>0.00</td>
<td>0</td>
<td>DYNAMIC SQL STMT TOTAL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>REOPTIMIZATION</td>
<td>0.00</td>
<td>0</td>
<td>NOT FOUND IN CACHE</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>NOT FOUND IN CACHE</td>
<td>0.00</td>
<td>0</td>
<td>FOUND IN CACHE</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>FOUND IN CACHE</td>
<td>0.00</td>
<td>0</td>
<td>IMPLICIT PREPARES</td>
<td>7.00</td>
<td>7</td>
</tr>
<tr>
<td>IMPLICIT PREPARES</td>
<td>0.00</td>
<td>0</td>
<td>PREPARES AVOIDED</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>PREPARES AVOIDED</td>
<td>0.00</td>
<td>0</td>
<td>CACHE LIMIT EXCEEDED</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>CACHE LIMIT EXCEEDED</td>
<td>0.00</td>
<td>0</td>
<td>PREP_STMT_PURGED</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>PREP_STMT_PURGED</td>
<td>0.00</td>
<td>0</td>
<td>CSWL - STMTS_PARSED</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>CSWL - STMTS_PARSED</td>
<td>0.00</td>
<td>0</td>
<td>CSWL - LITS_REPLACED</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>CSWL - LITS_REPLACED</td>
<td>0.00</td>
<td>0</td>
<td>CSWL - MATCHES_FOUND</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>CSWL - MATCHES_FOUND</td>
<td>0.00</td>
<td>0</td>
<td>CSWL - DUPLS_CREATED</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>CSWL - DUPLS_CREATED</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Trace:**

<table>
<thead>
<tr>
<th>Field</th>
<th>Average</th>
<th>Total</th>
<th>Field</th>
<th>Average</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>DYNAMIC SQL STMT AVERAGE</td>
<td></td>
<td></td>
<td>DYNAMIC SQL STMT TOTAL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>REOPTIMIZATION</td>
<td></td>
<td></td>
<td>NOT FOUND IN CACHE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NOT FOUND IN CACHE</td>
<td></td>
<td></td>
<td>FOUND IN CACHE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FOUND IN CACHE</td>
<td></td>
<td></td>
<td>IMPLICIT PREPARES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IMPLICIT PREPARES</td>
<td></td>
<td></td>
<td>PREPARES AVOIDED</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PREPARES AVOIDED</td>
<td></td>
<td></td>
<td>CACHE LIMIT EXCEEDED</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CACHE LIMIT EXCEEDED</td>
<td></td>
<td></td>
<td>PREP_STMT_PURGED</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PREP_STMT_PURGED</td>
<td></td>
<td></td>
<td>CSWL - STMTS_PARSED</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSWL - STMTS_PARSED</td>
<td></td>
<td></td>
<td>CSWL - LITS_REPLACED</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSWL - LITS_REPLACED</td>
<td></td>
<td></td>
<td>CSWL - MATCHES_FOUND</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSWL - MATCHES_FOUND</td>
<td></td>
<td></td>
<td>CSWL - DUPLS_CREATED</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSWL - DUPLS_CREATED</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**REOPTIMIZATION**

The total number of times reoptimization occurs because the value of the host variable or parameter marker changes.

Field Name: QXSTREOP

**NOT FOUND IN CACHE**

The number of times that DB2 searched the prepared statement cache but could not find a suitable prepared statement.

Field Name: QXSTNFND

**FOUND IN CACHE**

The number of times a PREPARE command was satisfied by copying a statement from the prepared statement cache.

Field Name: QXSTFND

**IMPLICIT PREPARES**

An implicit prepare occurs when the user copy of the prepared SQL statement no longer exists in the local dynamic SQL cache and the application plan or package is bound with KEEP_DYNAMIC YES.

If the skeleton copy of the prepared SQL statement exists in the global dynamic SQL cache in the EDM pool, a short prepare is executed, otherwise a full prepare is executed.

Field Name: QXSTIPRP

**PREPARES AVOIDED**
Dynamic SQL Statement

This field indicates the number of times where no SQL PREPARE or EXECUTE IMMEDIATE was issued by the application and a copy of a prepared SQL statement was found in local dynamic SQL cache.

When an application plan or package is bound with KEEPDYNAMIC YES, a copy of each prepared SQL statement for the application thread is held in the local dynamic SQL cache and kept across a commit boundary.

An application thread can save the total cost of a prepare by using a copy of the prepared statement in the local dynamic SQL cache from an earlier prepare by the same thread. To do this, the application must be modified to avoid issuing repetitive SQL PREPAREs for the same SQL statement.

Field Name: QXSTNPRP

CACHE_LIMIT_EXCEED
The number of times statements are invalidated in the local dynamic SQL cache because the MAXKEEPD limit has been reached and prepared SQL statements in the local dynamic SQL cache have to be reclaimed.

Field Name: QXSTDEXP

PREP_STMT_PURGED
The number of times statements are invalidated in the local dynamic SQL cache because of SQL DDL or updated RUNSTATS information and prepared SQL statements in the local dynamic SQL cache have to be reclaimed.

Field Name: QXSTDINV

CSWL - STMTS PARSED
The number of times DB2 parsed dynamic statements because CONCENTRATE STATEMENTS WITH LITERALS behavior was used for the prepare of the statement for the dynamic statement cache.

Field Name: QXSTCWLP

CSWL - LITS REPLACED
The number of times DB2 replaced at least one literal in a dynamic statement because CONCENTRATE STATEMENTS WITH LITERALS was used for the prepare of the statement for dynamic statement cache.

Field Name: QXSTCWLR

CSWL - MATCHES FOUND
The number of times DB2 found a matching reusable copy of a dynamic statement in cache during prepare of a statement that had literals replaced because of CONCENTRATE STATEMENTS WITH LITERALS.

Field Name: QXSTCWLM

CSWL - DUPLS CREATED
The number of times DB2 created a duplicate STMT instance in the statement cache for a dynamic statement that had literals replaced by CONCENTRATE STATEMENTS WITH LITERALS behavior. The duplicate STMT instance was needed because a cache match failed because the literal reusability criteria was not met.

Field Name: QXSTCWLD
Global Contention L-Locks

Global Contention L-Locks

This topic shows detailed information about “Accounting - Global Contention L-Locks”.

This block provides global contention information for a logical lock (L-lock) at plan level. It shows conflicts on locking requests between different DB2 members of a data sharing group when those members are trying to serialize shared resources.

For formatting reasons, OMEGAMON XE for DB2 PE shows different labels for report and trace. The following example shows both layouts, the report on the left, and the trace layout on the right.

Accounting - Global Contention L-Locks

The field labels shown in the following sample layout of “Accounting - Global Contention L-Locks” are described in the following section.

L-LOCKS - AVERAGE TIME/ELAPSED TIME

The accumulated global contention wait time for all L-locks.

Field Name: ADLKSUST

L-LOCKS - AV.EVENT/EVENTS

The number of global contention waits for all L-locks.

Field Name: ADLKSUSC

PARENT (DB,TS,TAB, PART) - AVERAGE TIME/ELAPSED TIME

The accumulated global contention wait time for parent L-locks.

A parent L-lock can be one of the following types:

• Database
• Tablespace
• Table
• Partition

Background and Tuning Information

Performance Expert might adjust this value if the thread was suspended when performance data was gathered.

Field Name: QWACAWTJ

PARENT (DB,TS,TAB, PART) - AV.EVENT/EVENTS

The number of global contention wait events for parent L-locks.

Field Name: ADLPSUSC

CHILD (PAGE, ROW) - AVERAGE TIME/ELAPSED TIME

The accumulated global contention wait time for child L-locks.

A child L-lock type can be:

• Page
Global Contention L-Locks

- Row
  Field Name: QWACAWTK

CHILD (PAGE,ROW) - AV.EVENT/EVENTS
The number of global contention wait events for child L-locks.
Field Name: ADLCSUSC

OTHER - AVERAGE TIME/ELAPSED TIME
The accumulated global contention wait time for other L-locks. Global extend lock is acquired in exclusive mode by Inserters before an extend service task switch.
Field Name: QWACAWTM

OTHER - AV.EVENT/EVENTS
The number of global contention wait events for other L-locks.
Field Name: ADLOSUSC
Global Contention P-Locks

Global Contention P-Locks
This topic shows detailed information about “Accounting - Global Contention P-Locks”.

This block provides global contention information for a physical lock (P-lock) at plan level. It shows conflicts on locking requests between different DB2 members of a data sharing group when those members are trying to serialize shared resources.

For formatting reasons, OMEGAMON XE for DB2 PE shows different labels for report and trace. The following example shows both layouts, the report on the left, and the trace layout on the right.

Accounting - Global Contention P-Locks

The field labels shown in the following sample layout of “Accounting - Global Contention P-Locks” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADPLSUST</td>
<td>P-LOCKS - AVERAGE TIME/ELAPSED TIME</td>
</tr>
<tr>
<td>ADPLSUSC</td>
<td>P-LOCKS - AVG.EVENT/EVENTS</td>
</tr>
<tr>
<td>QWACAWTN</td>
<td>PAGESET/PARTITION - AVERAGE TIME/ELAPSED TIME</td>
</tr>
<tr>
<td>ADPSSUSC</td>
<td>PAGESET/PARTITION - AVG.EVENT/EVENTS</td>
</tr>
<tr>
<td>QWACAWTO</td>
<td>PAGE - AVERAGE TIME/ELAPSED TIME</td>
</tr>
<tr>
<td>ADPPSUSC</td>
<td>PAGE - AVG.EVENT/EVENTS</td>
</tr>
<tr>
<td></td>
<td>OTHER - AVERAGE TIME/ELAPSED TIME</td>
</tr>
</tbody>
</table>

P-LOCKS - AVERAGE TIME/ELAPSED TIME
The accumulated global contention wait time for all P-locks.

Field Name: ADPLSUST

P-LOCKS - AVG.EVENT/EVENTS
The number of global contention waits for all P-locks.

Field Name: ADPLSUSC

PAGESET/PARTITION - AVERAGE TIME/ELAPSED TIME
The accumulated global contention time for pageset and partition P-locks.

Field Name: QWACAWTN

PAGESET/PARTITION - AVG.EVENT/EVENTS
The number of global contention waits for pageset and partition P-locks.

Field Name: ADPSSUSC

PAGE - AVERAGE TIME/ELAPSED TIME
The accumulated global contention wait time for page P-locks.

Field Name: QWACAWTO

PAGE - AVG.EVENT/EVENTS
The number of global contention waits for page P-locks.

Field Name: ADPPSUSC

OTHER - AVERAGE TIME/ELAPSED TIME
The accumulated global contention wait time for other P-locks. Includes suspension for Castout P-Locks and DBET locks. It could be because of Index Split processing which can be minimized if the Index key size is not large. If you can minimize the number of Index Keys in the Index, it will
Global Contention P-Locks

help to reduce the number of Index splits. For DB2 9, a large Index page size could also reduce the Index splits.

Field Name: QWACAWTQ

OTHER - AV.EVENT/EVENTS

The number of global contention waits for other P-locks.

Field Name: ADPOSUSC
Group Buffer Pool Activity

This topic shows detailed information about “Accounting - Group Buffer Pool Activity”. This block is printed for each active group buffer pool. When there is more than one active group buffer pool, a totals block is printed for each aggregation.

The following example shows both layouts, the report on the left, and the trace layout on the right.

Accounting - Group Buffer Pool Activity

The field labels shown in the following sample layout of “Accounting - Group Buffer Pool Activity” are described in the following section.

<table>
<thead>
<tr>
<th>Report:</th>
<th>Trace:</th>
</tr>
</thead>
<tbody>
<tr>
<td>GROUP BP1</td>
<td>AVERAGE</td>
</tr>
<tr>
<td>----------</td>
<td>----------</td>
</tr>
<tr>
<td>GBP-DEPEND GETPAGES</td>
<td>0.00</td>
</tr>
<tr>
<td>READ(XI)-DATA RETUR</td>
<td>0.00</td>
</tr>
<tr>
<td>READ(XI)-NO DATA RT</td>
<td>0.00</td>
</tr>
<tr>
<td>READ(NF)-DATA RETUR</td>
<td>0.00</td>
</tr>
<tr>
<td>READ(NF)-NO DATA RT</td>
<td>0.00</td>
</tr>
<tr>
<td>PREFETCH PAGES READ</td>
<td>0.00</td>
</tr>
<tr>
<td>CLEAN PAGES WRITTEN</td>
<td>0.00</td>
</tr>
<tr>
<td>UNREGISTER PAGE</td>
<td>0.00</td>
</tr>
<tr>
<td>ASYNCH GBP REQUESTS</td>
<td>0.00</td>
</tr>
<tr>
<td>EXPLICIT X-INVALID</td>
<td>0.00</td>
</tr>
<tr>
<td>ASYNCH SEC-GBP REQ</td>
<td>0.00</td>
</tr>
<tr>
<td>PG P-LOCK LOCK REQ</td>
<td>0.00</td>
</tr>
<tr>
<td>SPACE MAP PAGES</td>
<td>0.00</td>
</tr>
<tr>
<td>DATA PAGES</td>
<td>0.00</td>
</tr>
<tr>
<td>INDEX LEAF PAGES</td>
<td>0.00</td>
</tr>
<tr>
<td>PG P-LOCK UNLOCK REQ</td>
<td>0.00</td>
</tr>
<tr>
<td>PG P-LOCK LOCK SUSP</td>
<td>0.00</td>
</tr>
<tr>
<td>SPACE MAP PAGES</td>
<td>0.00</td>
</tr>
<tr>
<td>DATA PAGES</td>
<td>0.00</td>
</tr>
<tr>
<td>INDEX LEAF PAGES</td>
<td>0.00</td>
</tr>
<tr>
<td>WRITE AND REGISTER</td>
<td>0.00</td>
</tr>
<tr>
<td>WRITE &amp; REGISTER MULT</td>
<td>0.00</td>
</tr>
<tr>
<td>CHANGED PAGES WRITTEN</td>
<td>0.00</td>
</tr>
<tr>
<td>COMPL CHECKS SUSPEND</td>
<td>0.00</td>
</tr>
</tbody>
</table>

GBP-DEPEND GETPAGES
The number of Getpages made for GBP-dependent objects. This indicates the degree of data sharing.

Field Name: QBGAGG

READ(XI)-DATA RETUR
The number of coupling facility read requests required because the buffer was marked invalid. Data is returned from the group buffer pool.

Field Name: QBGAXD

READ(XI)-NO DATA RT
The number of synchronous coupling facility read requests necessary because the buffer was marked invalid. Data is not returned from the group buffer pool.

Field Name: ABGAXR
Group Buffer Pool Activity

This is an exception field.

READ(NF)-DATA RETUR
The number of coupling facility read requests necessary because the requested page was not found in the buffer pool. Data is returned from the coupling facility.

Field Name: QBGAMD
This is an exception field.

READ(NF)-NO DATA RT
The number of synchronous coupling facility read requests necessary because the requested page was not found in the buffer pool. Data is not returned from the coupling facility.

Field Name: ABGAMR
This is an exception field.

PREFETCH PAGES READ
The number of pages read from the group buffer pool due to prefetch under the control of the agent.

Field Name: QBGAMN
This is an exception field.

CLEAN PAGES WRITTEN
The number of clean pages written to the group buffer pool.

Field Name: QBGAWC
This is an exception field.

UNREGISTER PAGE
The number of coupling facility requests to unregister a page.

Field Name: QBGADG
This is an exception field.

ASYNCH GBP REQUESTS
The number of asynchronous IXLCACHE invocations for the primary group buffer pool.

Field Name: QBGAHS

EXPLICIT X-INVALID
The number of times an explicit coupling facility cross-invalidation request was issued.

Field Name: QBGAEX

ASYNCH SEC-GBP REQ
The number of IXLCACHE invocations for the secondary group buffer pool.

Field Name: QBGA2H

PG P-LOCK LOCK REQ
The number of all page P-lock lock requests.
Group Buffer Pool Activity

Field Name: ABGAPLR
SPACE MAP PAGES
The number of page P-lock lock requests for space-map pages.
Field Name: QBGAP1

DATA PAGES
The number of page P-lock lock requests for data pages.
Field Name: QBGAP2

INDEX LEAF PAGES
The number of page P-lock lock requests for index-leaf pages.
Field Name: QBGAP3

PG P-LOCK UNLOCK REQ
The number of page P-lock unlock requests.
Field Name: QBGAU1

PG P-LOCK LOCK SUSP
The sum of all page P-lock lock suspensions.
Field Name: ABGAPLS

SPACE MAP PAGES
The number of page P-lock suspensions for space-map pages.
Field Name: QBGAS1

DATA PAGES
The number of page P-lock suspensions for data pages.
Field Name: QBGAS2

INDEX LEAF PAGES
The number of page P-lock suspensions for index-leaf pages.
Field Name: QBGAS3

WRITE AND REGISTER
The number of Write and Register requests.
Field Name: QBGAWS

WRITE & REGISTER MULT
The number of Write and Register Multiple requests.
Field Name: QBGAWM

CHANGED PAGES WRITTEN
The number of changed pages written to the group buffer pool as a result of write and register (WAR), or write and register multiple (WARM) requests.
Field Name: QBGASW
This is an exception field.

WRITE TO SEC-GBP
**Group Buffer Pool Activity**

The number of requests to write changed pages to the secondary GBP for duplexing.

**Field Name:** QBGA2W

**COMPL CHECKS SUSPEND**

The number of completion checks for writes to the secondary GBP that were suspended because the write had not yet been completed.

**Field Name:** QBGA2S
Highlights

This topic shows the report and trace blocks for highlights. They present values such as the total number of threads and commitments for the entire group.

“Highlights - Report” on page 5-95
This topic shows detailed information about “Accounting - Highlights - Report”.

“Highlights - Trace” on page 5-101
This topic shows detailed information about “Accounting - Highlights - Trace”.

5-94 OMEGAMON XE for DB2 PE & PM: Report Reference
Highlights - Report

Accounting - Highlights - Report

The field labels shown in the following sample layout of “Accounting - Highlights - Report” are described in the following section.

### HIGHLIGHTS

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>#OCCURRENCES</td>
<td>13</td>
</tr>
<tr>
<td>#ALLIEDS</td>
<td>13</td>
</tr>
<tr>
<td>#ALLIEDS DISTRIB</td>
<td>0</td>
</tr>
<tr>
<td>#DBATS</td>
<td>0</td>
</tr>
<tr>
<td>#DBATS DISTRIB.</td>
<td>0</td>
</tr>
<tr>
<td>#NO PROGRAM DATA</td>
<td>13</td>
</tr>
<tr>
<td>#NORMAL TERMINAT</td>
<td>13</td>
</tr>
<tr>
<td>#DDFRRSAF ROLLUP</td>
<td>0</td>
</tr>
<tr>
<td>#ABNORMAL TERMIN</td>
<td>0</td>
</tr>
<tr>
<td>#CP/X PARALLEL.</td>
<td>0</td>
</tr>
<tr>
<td>#UTIL PARALLEL.</td>
<td>0</td>
</tr>
<tr>
<td>#IO PARALLELISM</td>
<td>0</td>
</tr>
<tr>
<td>#PCA RUP COUNT</td>
<td>0</td>
</tr>
<tr>
<td>#RUP AUTONOM. PR</td>
<td>0</td>
</tr>
<tr>
<td>#AUTONOMOUS PR</td>
<td>0</td>
</tr>
<tr>
<td>#INCREMENT. BIND</td>
<td>0</td>
</tr>
<tr>
<td>#COMMTS</td>
<td>215</td>
</tr>
<tr>
<td>#ROLLBACKS</td>
<td>0</td>
</tr>
<tr>
<td>#SVPT REQUESTS</td>
<td>0</td>
</tr>
<tr>
<td>#SVPT RELEASE</td>
<td>0</td>
</tr>
<tr>
<td>#SVPT ROLLBACK</td>
<td>0</td>
</tr>
<tr>
<td>MAX SQL CASL LVL.</td>
<td>0</td>
</tr>
<tr>
<td>UPDATE/COMMIT</td>
<td>0.00</td>
</tr>
<tr>
<td>SYNCH I/O AVG.</td>
<td>0.002932</td>
</tr>
</tbody>
</table>

**#OCCURRENCES**

The number of logical accounting records. A logical accounting record can contain more than one physical record.

This is the case, for example, in query CP and sysplex query parallelism, where several accounting records (IFCID 003 and, optionally, 239) are generated, namely one for the entire thread and one for each parallel task within the thread.

In case of Distributed Data Facility (DDF) or Recoverable Resource Manager Services Attach Facility (RRSAF) threads, it is the number of accounting intervals rolled up in a record.

This number is used for calculating averages (as a divisor) for class 1, 2, 3, and 5 times and events.

**Field Name:** ASOCCURS

**#ALLIEDS**

The number of allied threads. In case of Distributed Data Facility (DDF) or Recoverable Resource Manager Services Attach Facility (RRSAF) threads, it is the number of accounting intervals rolled up in this record for the corresponding end user.

**Field Name:** ASALLIED

**#ALLIEDS DISTRIB**
The number of allied-distributed threads. In case of Distributed Data Facility (DDF) or Recoverable Resource Manager Services Attach Facility (RRSAF) threads, it is the number of accounting intervals rolled up in this record for the corresponding end user.

Field Name: ASALLDST

#DBATS

The number of database access threads. In case of Distributed Data Facility (DDF) or Recoverable Resource Manager Services Attach Facility (RRSAF) threads, it is the number of accounting intervals rolled up in this record for the corresponding end user.

Field Name: ASDBATS

#DBATS DISTRIB.

The number of DBAT-distributed threads. In case of Distributed Data Facility (DDF) or Recoverable Resource Manager Services Attach Facility (RRSAF) threads, it is the number of accounting intervals rolled up in this record for the corresponding end user.

Field Name: ASDBATD

#NO PROGRAM DATA

The number of Accounting records without package data. In case of Distributed Data Facility (DDF) or Recoverable Resource Manager Services Attach Facility (RRSAF) threads, it cannot be determined. In this case it is 0.

Field Name: ADNOPACK

#NORMAL TERMINAT

The number of normal terminations. Here is a list of reasons for termination and the corresponding field names:

<table>
<thead>
<tr>
<th>Reason</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>New user</td>
<td>ASNTNEWU</td>
</tr>
<tr>
<td>Dealllocation</td>
<td>ASNTDEAL</td>
</tr>
<tr>
<td>Application program end</td>
<td>ASNTAPEN</td>
</tr>
<tr>
<td>Resignon</td>
<td>ASNTRESI</td>
</tr>
<tr>
<td>DBAT inactive</td>
<td>ASNTDBAT</td>
</tr>
<tr>
<td>RRS commit</td>
<td>ASRRRCOM</td>
</tr>
</tbody>
</table>

Note: Termination reasons in case of Distributed Data Facility (DDF) or Recoverable Resource Manager Services Attach Facility (RRSAF) threads are not counted.

Field Name: ASNORMTM
#DDFRRSAF ROLLUP

The number of DDF/RRSAF rollup records with accumulated counter data for an end user.

A rollup record is written when the number of occurrences of the end user on the thread reaches the ZPARM value for ACCUMACC and due to one of the following reasons:

- The number of times the threshold was reached for the number of end-user occurrences when data was accumulated by end user for DDF or RRSAF.
- The number of times the DB2 storage threshold for Accounting blocks was reached for data accumulated by end user for DDF or RRSAF.
- The number of times the threshold for the staleness was exceeded when data was accumulated by end user for DDF or RRSAF.

Note: End user is defined as the concatenation of the following values:

- End-user user ID (QWHEUID, 16 bytes)
- End-user transaction name (QWHCEUTX, 32 bytes)
- End-user workstation name (QWHCEUWN, 18 bytes)

Field Name: ASCUTS

#ABNORMAL TERMIN

The number of abnormal terminations. Here is a list of reasons for termination and the corresponding field names:

<table>
<thead>
<tr>
<th>Reason</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application program abend</td>
<td>ASATAPAB</td>
</tr>
<tr>
<td>End of memory</td>
<td>ASATENDM</td>
</tr>
<tr>
<td>Resolve indoubt</td>
<td>ASATRIND</td>
</tr>
<tr>
<td>Cancel force</td>
<td>ASATCANF</td>
</tr>
</tbody>
</table>

Field Name: ASABNOTM

This is an exception field.

#CP/X PARALLEL

The number of originating accounting records where query CP and sysplex query parallelism was used for at least one SQL statement. I/O parallelism might have been used by other SQL statements.

Field Name: ASPARCPU

#UTIL PARALLEL

The number of Accounting records that indicated that UTILITY parallelism was used by at least one SQL statement and query CP and sysplex query parallelism was not used by any SQL statement.

Field Name: ASPARUT
# IO PARALLELISM

The number of accounting records that indicated that I/O parallelism was used by at least one SQL statement and query CP and sysplex query parallelism was not used by any SQL statement.

Field Name: ASPARIO

#PCA RUP COUNT

The number of parallel child agents rolled into this record. The value depends on the record type:
1. For all non-rollup records, this value is 0.
2. For a parallel query rollup record, this value is the number of parallel child agents rolled into this record.
3. For a DDF/RRSAF rollup record, this value is the number of parallel query child agents rolled into this record. These agents are NOT counted in QWACPCNT.
4. For an autonomous procedure rollup record, this value is 0.

Field Name: APTCOUNT

# RUP AUTONOM. PR

The number of accounting records that indicated a roll-up autonomous thread.

Field Name: ADRUPATX

#AUTONOMOUS PR

The value depends on the record type:
1. For non-rollup records, this value is the number of autonomous transactions executed.
2. For parallel query rollup records, this value is 0.
3. For a DDF/RRSAF rollup records, this value is the number of autonomous transactions executed. These transactions are NOT counted in QWACPCNT.
4. For autonomous transaction rollup records, this value is 0.

Field Name: AATCOUNT

# INCREMENT. BIND

The number of incremental binds (excluding prepare). It is incremented by:
- SQL statements with BIND VALIDATE(RUN) that fail at bind time and are bound again at execution time
- Static DDL statements (such as CREATE TABLE, DROP TABLE, LOCK TABLE) that use DB2 private protocol

Background and Tuning Information

If a plan is bound with VALIDATE(RUN), DB2 performs validity checks at bind time and rechecks any failures at run time. This can result in catalog contention and degraded application performance, depending on the number of statements flagged and how many times they are executed. Avoid VALIDATE(RUN) if possible. Ensure that all objects are created and all privileges are granted before bind, and select the VALIDATE(BIND) option.

Field Name: QXINCRB
This is an exception field.

#COMMISSIONS
The number of successful two-phase (units of recovery) or single-phase (syncs) commit requests. It indicates the number of units of recovery that are completed successfully, and for which the associated commit duration locks were released. It represents the total number of commit requests processed by the DB2 subsystem, whether the request was an explicit or implicit external request from an IMS or a CICS connection, or an implicit internal request within DB2 when DB2 was the commit coordinator or conducted read-only commit processing as a commit participant on phase-1 calls from an IMS or CICS connection.

For parallel queries, only the commits from the initiating (parent) thread are recorded by this counter.

Field Name: QWACCOMM
This is an exception field.

#ROLLBACKS
The number of rollback requests. This is the number of units that were backed out, including rollbacks from attache.

Special Considerations: This field contains the number of:
- Application program abends
- Application rollback requests
- Application deadlocks on database records
- Applications canceled by operator
- Thread abends due to resource shortage

Field Name: QWACABRT
This is an exception field.

#SVPT REQUESTS
The number of named SAVEPOINTs set within a transaction.

Field Name: QWACSVPT

#SVPT RELEASE
The number of RELEASE SAVEPOINT statements executed.

Background and Tuning Information
Release savepoints as soon as possible. Outstanding savepoints block SQL operations that resolve remote locations. DB2 always releases outstanding savepoints when a transaction ends.

Field Name: QWACRLSV
This is an exception field.

#SVPT ROLLBACK
The number of ROLLBACK TO SAVEPOINT statements executed.

Field Name: QWACRBSV

MAX SQL CASC LVL
The maximum level of indirect SQL cascading. This includes cascading because of triggers, UDFs, or stored procedures.
Field Name: QXCASCDP
This is an exception field.

UPDATE/COMMIT
The sum of SQL INSERT, SQL UPDATE, and SQL DELETE statements executed.

Field Name: ASIUD
This is an exception field.

SYNCH I/O AVG.
The synchronous I/O suspension time per event.

Field Name: AAIOTMCN
This is an exception field.
**Highlights - Trace**

This topic shows detailed information about “Accounting - Highlights - Trace”.

**Accounting - Highlights - Trace**

The field labels shown in the following sample layout of “Accounting - Highlights - Trace” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>THREAD TYPE</td>
<td>ALLIED: The thread is not involved in any distributed activity. ALLEDDIST: The thread is initiated by a DB2 attach and requests data from one or more server locations. DBAT: The thread is initiated, created, and performing work on behalf of a remote (requester) location. The value DBAT also includes DBAT DISTRIBUTED threads that are initiated by a requester location and executed by the server location that in turn requests data from another server location.</td>
</tr>
<tr>
<td>TERM.CONDITION</td>
<td>Signon in a CICS environment is controlled by an additional RCT option, TXIDSO. If YES, resignon occurs if the only identifier changed is the transaction ID. If NO, resignon does not occur.</td>
</tr>
<tr>
<td>INVOKE REASON</td>
<td>The status of the thread. The values are:</td>
</tr>
<tr>
<td>PARALLELISM</td>
<td></td>
</tr>
<tr>
<td>PCA RUP COUNT</td>
<td></td>
</tr>
<tr>
<td>RUP AUTONOMOUS PR</td>
<td></td>
</tr>
<tr>
<td>AUTONOMOUS PR</td>
<td></td>
</tr>
<tr>
<td>QUANTITY</td>
<td></td>
</tr>
<tr>
<td>COMMITS</td>
<td></td>
</tr>
<tr>
<td>ROLLBACK</td>
<td></td>
</tr>
<tr>
<td>SVPT REQUESTS</td>
<td></td>
</tr>
<tr>
<td>SVPT RELEASE</td>
<td></td>
</tr>
<tr>
<td>SVPT ROLLBACK</td>
<td></td>
</tr>
<tr>
<td>INCREM.BINDS</td>
<td></td>
</tr>
<tr>
<td>UPDATE/COMMIT</td>
<td></td>
</tr>
<tr>
<td>SYNC I/O AVG.</td>
<td></td>
</tr>
<tr>
<td>PROGRAMS</td>
<td></td>
</tr>
<tr>
<td>MAX CASCADE</td>
<td></td>
</tr>
</tbody>
</table>

**Background and Tuning Information**

If the thread is involved in distributed activity, some monitored values can produce different results. For example, the class 1 elapsed time for a distributed thread is higher because VTAM time is also included.

Field Name: ADTHRTYP

Field Name: ADTERMCO

Field Name: ADTERMSO
CAN FORCE
   CANCEL FORCE. The Stop Force command terminated, abnormal program termination.

DBAT INACT
   DDF thread is becoming inactive.

DEALLOC
   Deallocation, normal program termination.

TYP2 INACT
   DDF TYPE 2 thread is becoming inactive.

MEMORY END
   End of memory, abnormal termination.

MON READS
   IFI reads request for IFCID 147.

NEW USER
   New user, the authorization Id changed.

PROG ABEND
   End of task. Application program abended.

PROGRM END
   End of task. Application program terminated normally.

RESIGNON
   Same user resign-on with same authorization ID.

RES INDBT
   Resolve indoubt, abnormal program termination.

RRS COMMIT
   Termination due to a commit of an application attached to the Recoverable Resource Manager Services Facility (RRSAF).

STALENESS
   Accumulating data by end user for DDF or RRSAF and accumulated data has exceeded the staleness threshold.

BLOCK STOR
   Accumulating data by end user for DDF or RRSAF and internal DB2 storage threshold has been reached.

TASK END
   End of task - application program terminated normally.

END USER
   Accumulating data by end user for DDF or RRSAF and threshold reached for number end user occurrences.

Field Name: ADINVRSN

PARALLELISM
   An indicator to show which type of parallel processing is used when SQL statements are executed:

   SQL statement
      Parallel processing

   I/O     For threads exploiting query I/O parallelism but no query CP or sysplex query parallelism
CP    For threads exploiting query CP parallelism

SYSPLEX
   For threads exploiting sysplex query parallelism

UTILITY
   For utility threads with subtasks

NO    For threads without subtasks

Field Name: ADPARLEV

PCA RUP COUNT

The number of parallel child agents rolled into this record. The value
depends on the record type:
1. For all non-rollup records, this value is 0.
2. For a parallel query rollup record, this value is the number of parallel
   child agents rolled into this record.
3. For a DDF/RRSAF rollup record, this value is the number of parallel
   query child agents rolled into this record. These agents are NOT
   counted in QWACPCNT.
4. For autonomous procedure rollup record, this value is 0.

Field Name: APTCOUNT

RUP AUTONOM.PR

The number of accounting records that indicated a roll-up autonomous
thread.

Field Name: ADRUPATX

AUTONOMOUS PR

The value depends on the record type:
1. For non-rollup records, this value is the number of autonomous
   transactions executed.
2. For parallel query rollup records, this value is 0.
3. For a DDF/RRSAF rollup records, this value is the number of
   autonomous transactions executed. These transactions are NOT counted
   in QWACPCNT.
4. For autonomous transaction rollup records, this value is 0.

Field Name: AATCOUNT

QUANTITY

The number of parallel child agents, or Accounting intervals rolled up, or
autonomous procedures rolled up. The value depends on the record type:
• For a non-rollup parent record, this value is the number of parallel child
  agents that were created.
• For a non-rollup child agent record, this value is 0.
• For a parallel query rollup record, this value is the number of parallel
  child agents rolled into the record.
• For a DDF/RRSAF rollup record, this value is the number of Accounting
  intervals that were rolled into the record for the corresponding end user.
• For an autonomous procedure rollup record, this value is the number of
  autonomous procedures rolled into the record.

Field Name: QWACPCNT
COMMENTS

The number of successful two-phase (units of recovery) or single-phase (syncs) commit requests. It indicates the number of units of recovery that are completed successfully, and for which the associated commit duration locks were released. It represents the total number of commit requests processed by the DB2 subsystem, whether the request was an explicit or implicit external request from an IMS or a CICS connection, or an implicit internal request within DB2 when DB2 was the commit coordinator or conducted read-only commit processing as a commit participant on phase-1 calls from an IMS or CICS connection.

For parallel queries, only the commits from the initiating (parent) thread are recorded by this counter.

Field Name: QWACCOMM
This is an exception field.

ROLLBACK

The number of rollback requests. This is the number of units that were backed out, including rollbacks from attaches.

Special Considerations: This field contains the number of:
• Application program abends
• Application rollback requests
• Application deadlocks on database records
• Applications canceled by operator
• Thread abends due to resource shortage

Field Name: QWACABRT
This is an exception field.

SVPT REQUESTS

The number of named SAVEPOINTs set within a transaction.

Field Name: QWACSVPT

SVPT RELEASE

The number of RELEASE SAVEPOINT statements executed.

Background and Tuning Information
Release savepoints as soon as possible. Outstanding savepoints block SQL operations that resolve remote locations. DB2 always releases outstanding savepoints when a transaction ends.

Field Name: QWACRLSV
This is an exception field.

SVPT ROLLBACK

The number of ROLLBACK TO SAVEPOINT statements executed.

Field Name: QWACRBSV

INCREM.BINDS

The number of incremental binds (excluding prepare). It is incremented by:
• SQL statements with BIND VALIDATE(RUN) that fail at bind time and are bound again at execution time
Highlights - Trace

- Static DDL statements (such as CREATE TABLE, DROP TABLE, LOCK TABLE) that use DB2 private protocol

**Background and Tuning Information**

If a plan is bound with VALIDATE(RUN), DB2 performs validity checks at bind time and rechecks any failures at run time. This can result in catalog contention and degraded application performance, depending on the number of statements flagged and how many times they are executed. Avoid VALIDATE(RUN) if possible. Ensure that all objects are created and all privileges are granted before bind, and select the VALIDATE(BIND) option.

**Field Name:** QXINCRB

This is an *exception* field.

**UPDATE/COMMIT**

The sum of SQL INSERT, SQL UPDATE, and SQL DELETE statements executed.

**Field Name:** ASIUD

This is an *exception* field.

**SYNCH I/O AVG.**

The synchronous I/O suspension time per event.

**Field Name:** AAIOTMCN

This is an *exception* field.

**PROGRAMS**

The number of packages or DBRMs for which accounting data was collected.

**Field Name:** QWACPKGN

This is an *exception* field.

**MAX CASCADE**

The maximum level of indirect SQL cascading. This includes cascading because of triggers, UDFs, or stored procedures.

**Field Name:** QXCASCDDP

This is an *exception* field.
Identification

This topic shows detailed information about “Accounting - Identification”.

This block is shown for the accounting trace. It displays OMEGAMON XE for DB2 PE identifiers present in an accounting trace. These identifiers can be used on the ORDER option for an accounting report.

Accounting - Identification

The field labels shown in the following sample layout of “Accounting - Identification” are described in the following section.

--- IDENTIFICATION -----------------------------------------------
ACCT TSTAMP: 02/14/13 10:20:09.30 PLANNAME: IS-255-0 WLM SCL: STCMD
BEGIN TIME : 02/14/13 10:20:01.14 PROD TYP: JDBC DRIVER CICS NET: N/A
END TIME : N/P PROD VER: V3 R66M0 LUW NET: G99A83BD CICS INS: N/A
REQUESTER : ::FFFF:9.154.1#1 CORRNAME: db2jcc_a LUW LUN: GAC7
MAINPACK : IS-255-0 CORRNMBR: ppli LUW INS: CAEBCE3316A8
PRIMAUTH : MTS CONNECT : SERVER WSNAME : IS-255-012345678#1
ORIGAUTH : MTS CONNTYPE: DRDA LUW SEQ: 2 TRANSACT: IS-255-01234567890123456789012#1

ACCT TSTAMP
The store clock value of the time when the accounting record was generated.
Field Name: QWHSTCK

PLANNANE
The plan name. It is blank for a DB2 command thread; otherwise:

DSNESPRR
For SPUFI with repeatable read.

DSNESPBC
For SPUFI with cursor stability.

DSNUTIL
For utilities.

DSNTEP2
For DSNTEP2.

DSNBIND
For binding.

The application plan name
For IMS.

The application plan name
For CICS.

A blank plan name
For IMS and CICS commands.

DSQPLAN
For QMF.

The first 8 bytes of the application name
For DRDA connections to the common servers.

Field Name: QWHCPLAN
This is an exception field.

WLM SCL
Identification

The MVS™ workload manager service class name. This field is used for database access threads on MVS 5.2 or later.

Field Name: QWACWLME

CICS NET

The network ID of the accounting correlation token used to correlate DB2 IFC records to CICS records for the CICS transaction.

Field Name: ADCICSN

BEGIN TIME

The beginning store clock value for the period covered by the accounting record. You can determine the elapsed time of the application by subtracting this field from the ending store clock value (QWACESC). Threads that do not terminate (such as CICS primed threads and IMS wait-for-input message regions) can have an ending clock value that includes the time during which the thread was inactive and waiting for work.

If a roll-up trace record is written with accumulated counter data, QWACBSC represents the earliest begin store clock value for a thread that has rolled data into the record. In this case, QWACESC shows the accumulated elapsed time.

Field Name: QWACBSC

PROD TYP

Shows the product identifier (ID) of the requester. It can have the following values:

- **DB2** For DB2 UDB for z/OS
- **SQL/DS** For DB2 UDB for VSE and VM
- **JDBC DRIVER** For Universal JDBC driver
- **COMMON SERV** For DB2 UDB for Linux, UNIX, Windows
- **DB2/400** For DB2 UDB for iSeries®

Otherwise, it shows the first 3 characters of the product ID, or N/P if the record was written at the application requester location.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup. For parallel query rollup records, the value will be derived from the parent record.

Field Name: QWHDPRID

CICS LUN

The LU name of the accounting correlation token used to correlate DB2 IFC records to CICS records for the CICS transaction.

Field Name: ADCICSLU

END TIME
Identification

The ending store clock value. You can use this field with the beginning store clock value (QWACBSC) to determine the elapsed time of an application.

If a roll-up record is written with accumulated accounting data, QWACESC contains the accumulated elapsed time. In Accounting Trace reports, the elapsed time is shown under CLASS 1: NONNESTED ELAPSED TIME and the END TIME is reported as N/P, because QWACESC does not contain a timestamp. In the Accounting FILE GENERAL table, the accumulated elapsed time QWACESC is stored in column CLASS1_ELAPSED and column CLASS1_TIME_END contains a timestamp 1900-01-01-00.00.00.000000.

Field Name: QWACESC

PROD VER

The version, release, and modification level of the product, which generated the accounting information. It has the following format:

\[ Vv \]  Version level
\[ Rr \]  Release level
\[ Mm \]  Modification level

N/P is shown if the Product Type is not present or the record was written at the application requester location.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup. For parallel query rollup records, the value will be derived from the parent record.

Field Name: QWHDPRIDV

LUW NET

The network ID.

Field Name: QWHSNID

CICS INS

The instance number of the accounting correlation token.

Background and Tuning Information

The accounting correlation token is made up from the CICS Token Network ID, Token LU name, and instance number.

CICS generates an LU 6.2 unit of work ID for every CICS task, whether terminal or non-terminal driven.

If TOKENE=YES in the RCT entry, then the CICS logical unit of work ID (LUWID) less the commit count (2 bytes) is passed into this field. The first eight bytes are the network name. For CICS, this is a variable-length field, so the first eight bytes are right padded with blanks. The second eight bytes give the LU name. This is also a variable-length field in CICS and is, therefore, also right padded with blanks, as necessary. The final six bytes are the uniqueness value.

Field Name: ADCICSIN

REQUESTER

The location name of the requester. If the thread is an allied thread (no distributed requests) or the thread is an allied-distributed thread (this
location is the requester), OMEGAMON XE for DB2 PE sets this field equal to the local location. If the thread is a database access thread (this location is a server).

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup. For parallel query rollup records, the value will be derived from the parent record.

Field Name: QWHDRQNM

CORRNAME

This field shows the correlation name. It is obtained by translating the correlation ID into correlation name and number. The default translation depends on the connection type of the thread:

- **Batch**  Job name
- **TSO or CAF**  Original authorization ID
- **CICS**  Transaction ID
- **IMS**  Application PST
- **RRSAF**
  Characters 1 to 8 of the parameter correlation ID specified for SIGNON.

You can define your own correlation ID translation, which overrides the default translation.

Field Name: ADCORNME

LUW LUN

The name of the logical unit.

Field Name: QWHSLUNM

MAINPACK

The MAINPACK value, which is derived from a package name. If this is not possible (for example, if there are no QPAC sections), the MAINPACK value is initialized to the plan name.

Field Name: ADMAINPK

This is an exception field.

CORRNMBR

This field shows the correlation number. It is obtained by translating the correlation ID into correlation name and number. The default translation depends on the connection type of the thread:

- **Batch**  Blank
- **TSO or CAF**  Blank
- **CICS**  Pool thread
- **IMS**  Application PSBNAME
- **RRSAF**
  Characters 9 - 12 of the parameter correlation ID specified for SIGNON.
Identification

You can define your own correlation ID translation which overrides the default translation.

Field Name: ADCORNMB

LUW INS

The instance number. When concatenated with the fully qualified network name, it uniquely identifies a distributed thread.

Field Name: QWHSLUUV

ENDUSER

The user ID of the workstation end user. This user ID can be different from the authorization ID used to connect to DB2. This field contains blanks if the client does not supply this information.

Field Name: QWHCEUID

PRIMAUTH

The primary authorization ID from a connection or signon. The connection authorization exit and the signon authorization exit can change the primary authorization ID so that it differs from the original primary authorization ID (ORIGAUTH). Distributed authorization ID translation can also change the primary authorization ID.

Field Name: QWHCAID

CONNTYPE

The connecting system type code (in hexadecimal). This field can have a null value. Utilities, for example, do not have a connecting system type.

Field Name: QWHCATYP

LUW SEQ

The LUW sequence number, which identifies the last commit scope that the logical unit participated in. This number is incremented whenever a thread is committed or rolled back.

Field Name: QWHSLUCC

TRANSACT

The transaction or application name that is run.

Field Name: QWHCEUTX

ORIGAUTH

The original authorization ID. Possible values are:

- For TSO: the logon ID
- For batch: the user ID on the job statement
- For IMS (message-driven regions): the signon ID, LTERM, ASXBUSR, or PSB name
- For IMS (control regions): the user ID on the job statement, or the RACF® started procedure entry if RACF is used
- For CICS: the user ID, TERM ID, TRAN ID, or as specified in the resource control table
- For MVS operator commands and DB2 system internal agents: SYSOPR
- For a distributed application server (AS):
Identification

- If the application requester (AR) is a DB2 system, then this is the same value that was assigned at the AR.
- If the application requester is not a DB2 system, then this is the user ID used to make the initial connection with the application server.

**Field Name:** QWHCOPID

**CONNECT**

The connection name. Possible values are:
- For batch: BATCH
- For TSO: TSO
- For QMF: DB2CALL
- For utilities: UTILITY
- For DB2 private protocol this is the DB2 subsystem ID
- For IMS: the IMS ID
- For CICS, this is the CICS ID
- For DRDA connections from non-DB2 requesters: SERVER

**Field Name:** QWHCCN

This is an *exception* field.

**WSNAME**

The end user's workstation name.

**Field Name:** QWHCEUWN
Initial DB2 Common Server Traces

Initial DB2 Common Server Traces

This topic shows the trace blocks for the initial DB2 common server.

"Initial DB2 Common Server Correlation" on page 5-113
This topic shows detailed information about “Accounting - Initial DB2 Common Server Correlation”.

"Initial DB2 Common Server or Universal JDBC Driver Correlation” on page 5-115
This topic shows detailed information about “Accounting - Initial DB2 Common Server or Universal JDBC Driver Correlation”.
Initial DB2 Common Server Correlation

Initial DB2 Common Server Correlation
This topic shows detailed information about “Accounting - Initial DB2 Common Server Correlation”.

This block shows the Accounting trace for the initial DB2 common server correlation.

Accounting - Initial DB2 Common Server Correlation

The field labels shown in the following sample layout of “Accounting - Initial DB2 Common Server Correlation” are described in the following section.

---- INITIAL DB2 COMMON SERVER CORRELATION ----------------------------------------
PRODUCT ID : COMMON SERV
PRODUCT VERSION: V9 R1 M1
CLIENT PLATFORM: AIX 64BIT
CLIENT APPLNAME: PRCCHNG
CLIENT AUTHID : USRT001
DDCS ACC.SUFFIX: “BLANK”

PRODUCT ID
The product identifier of the requester:

DB2   DB2 UDB for z/OS
SQL/DS  DB2 UDB for VSE and VM
COMMON SERV  DB2 UDB for Linux, UNIX, and Windows
DB2/400  DB2 UDB for iSeries
N/P   If the record was written at the application requester location
Otherwise, it shows the first 3 characters of the product ID.
For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.
Field Name: QMDAPRIDP

PRODUCT VERSION
The version, release, and modification level of the product, which generated the accounting information. It has the following format: vvv rr m, where:

vvv  Version level
rr   Release level
m    Modification level
N/P is shown if the record was written at the application requester location.
For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.
Field Name: QMDAPRIDV

CLIENT PLATFORM
The client platform, such as AIX®. This is a 1 to 18 character field padded with blanks.
**Initial DB2 Common Server Correlation**

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

**Field Name:** QMDAPLAT

**CLIENT APPLNAME**

The name of the client application. This is a 1 to 20 character field padded with blanks. An example is "PAYROLL".

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

**Field Name:** QMDAAPPL

**CLIENT AUTHID**

The client authorization ID of an application process. This is a 1 to 8 character field padded with blanks. An example is "SMITH".

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

**Field Name:** QMDAATID

**DDCS ACC.SUFFIX**

The account suffix. The maximum length of this field is 200 bytes. This field is the user-supplied portion (suffix) of the accounting string. An example is "DEFAULT_DRDA". A value of zero in QMDASFLN Indicates there is no account suffix.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

**Field Name:** QMDASUFX
Initial DB2 Common Server or Universal JDBC Driver Correlation

This topic shows detailed information about “Accounting - Initial DB2 Common Server or Universal JDBC Driver Correlation”.

This block shows the accounting trace for the initial DB2 common server or universal JDBC driver correlation.

Accounting - Initial DB2 Common Server or Universal JDBC Driver Correlation

The field labels shown in the following sample layout of “Accounting - Initial DB2 Common Server or Universal JDBC Driver Correlation” are described in the following section.

<table>
<thead>
<tr>
<th>PRODUCT ID</th>
<th>PRODUCT VERSION</th>
<th>CLIENT PLATFORM</th>
<th>CLIENT APPLNAME</th>
<th>CLIENT AUTHID</th>
<th>DDCS ACC.SUFFIX</th>
</tr>
</thead>
<tbody>
<tr>
<td>JDBC DRIVER</td>
<td>V3 R66M0</td>
<td>IS-255-01234567890</td>
<td>IS-255-0123456789012</td>
<td>IS-128--</td>
<td>IS-255-01234567890123456789012#1</td>
</tr>
</tbody>
</table>

PRODUCT TYP

Shows the product identifier (ID) of the requester. It can have the following values:

- **DB2** For DB2 UDB for z/OS
- **SQL/DS** For DB2 UDB for VSE and VM
- **JDBC DRIVER** For Universal JDBC driver
- **COMMON SERV** For DB2 UDB for Linux, UNIX, Windows
- **DB2/400** For DB2 UDB for iSeries

Otherwise, it shows the first 3 characters of the product ID, or N/P if the record was written at the application requester location.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

Field Name: QMDAPRID

PRODUCT VERSION

The version, release, and modification level of the product, which generated the accounting information. It has the following format: \( vv \ rr \ m \), where:

- **\( vv \)** Version level
- **\( rr \)** Release level
- **\( m \)** Modification level

N/P is shown if the record was written at the application requester location.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

Field Name: QMDAPRIDV
CLIENT PLATFORM

The client platform, such as AIX. This is a 1 to 18 character field padded with blanks.
For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.
Field Name: QMDAPLAT

CLIENT APPLNAME

The name of the client application. This is a 1 to 20 character field padded with blanks. An example is "PAYROLL".
For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.
Field Name: QMDAAPPL

CLIENT AUTHID

The client authorization ID of an application process. This is a 1 to 8 character field padded with blanks. An example is "SMITH".
For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.
Field Name: QMDAATID

DDCS ACC.SUFFIX

The account suffix. The maximum length of this field is 200 bytes. This field is the user-supplied portion (suffix) of the accounting string. An example is "DEFAULT_DRDA". A value of zero in QMDASFLN Indicates there is no account suffix.
For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.
Field Name: QMDASUFX
Initial DB2 Requester Correlation

This topic shows detailed information about “Accounting - Initial DB2 Requester Correlation”.

This block shows the accounting trace for the initial DB2 requester correlation.

**Accounting - Initial DB2 Requester Correlation**

The field labels shown in the following sample layout of “Accounting - Initial DB2 Requester Correlation” are described in the following section.

```
---- INITIAL DB2 REQUESTER CORRELATION
PRODUCT ID : DB2
PRODUCT VERSION: CCCCCCCC
LOCATION NAME : CCCCCCCCCCCCCCCC
NET ID : CCCCCCCC
LU NAME : CCCCCCCC
AUTHID : CCCCCCCC
CONNTYPE : CCCCCCCC
CORRNAME : CCCCCCCC
CORRNMBR : CCCCCCCC
```

**PRODUCT ID**

Shows the product identifier (ID) of the requester. It can have the following values:

- **DB2** For DB2 UDB for z/OS
- **SQL/DS** For DB2 UDB for VSE and VM
- **JDBC DRIVER** For Universal JDBC driver
- **COMMON SERV** For DB2 UDB for Linux, UNIX, Windows
- **DB2/400** For DB2 UDB for iSeries

Otherwise, it shows the first 3 characters of the product ID, or N/P if the record was written at the application requester location.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

**Field Name:** QMDAPRID

**PRODUCT VERSION**

The version, release, and modification level of the product, which generated the accounting information. It has the following format: \( vv \ rr \ m \), where:

- \( vv \) Version level
- \( rr \) Release level
- \( m \) Modification level

N/P is shown if the record was written at the application requester location.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

**Field Name:** QMDBG2PRDV
Initial DB2 Requester Correlation

**LOCATION NAME**

The location name for the DB2 subsystem that created the QMDAINFO values.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

*Field Name: QMDALOCN*

**NET ID**

The NETID of the DB2 subsystem that created the QMDAINFO values.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

*Field Name: QMDANETN*

**LU NAME**

The SNA LU name of the DB2 subsystem that created the QMDAINFO values.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

*Field Name: QMDALUNM*

**AUTHID**

The DB2 authorization ID that the SQL application used before name translation and before driving the connection exit at the DB2 site where the SQL application is running.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

*Field Name: QMDAAUTH*

**CONNTYPE**

The type of subsystem connection at the DB2 system where the SQL application is running. Possible values and their descriptions are:

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BATCH</td>
<td>TSO or call attach</td>
</tr>
<tr>
<td>SASS</td>
<td>CICS</td>
</tr>
<tr>
<td>MASS</td>
<td>IMS</td>
</tr>
<tr>
<td>DIST</td>
<td>Distributed</td>
</tr>
</tbody>
</table>

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

*Field Name: QMDACTYP*

**CORRNAME**

The translated correlation name derived from the correlation ID. The translation depends on the connection ID.

*Field Name: ADRQCRNM*

**CORRNMBR**

The translated correlation number derived from the correlation ID. The translation depends on the connection ID.
Initial DB2 Requester Correlation

Field Name: ADRQCRNB
Initial Other Requester Correlation

Initial Other Requester Correlation

This topic shows detailed information about “Accounting - Initial Other Requester Correlation”.

This block shows the accounting trace for the initial other requester correlation.

Accounting - Initial Other Requester Correlation

The field labels shown in the following sample layout of “Accounting - Initial Other Requester Correlation” are described in the following section.

--- INITIAL OTHER REQUESTER CORRELATION

PRODUCT ID : SQL/DS
PRODUCT VERSION: CCCCCCCC
STRING : CCCCCCCC

PRODUCT ID

The product identifier of the requester:

DB2     DB2 UDB for z/OS
SQL/DS   DB2 UDB for VSE and VM
COMMON SERV  DB2 UDB for Linux, UNIX, and Windows
DB2/400  DB2 UDB for iSeries
N/P     If the record was written at the application requester location
        Otherwise, it shows the first 3 characters of the product ID.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

Field Name: QMDAPRIDP

PRODUCT VERSION

The version, release, and modification level of the product, which generated the accounting information. It has the following format: \( vv \ rr \ m \),

where:

\( vv \)  Version level
\( rr \)  Release level
\( m \)  Modification level

N/P is shown if the record was written at the application requester location.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

Field Name: QMDAPRIDV

STRING

The accounting string:

- For local DB2 threads, the format of the accounting string is shown in QMDAINFO.
- For database access threads, the accounting string contains the accounting string sent by the requester.
Initial Other Requester Correlation

- The QMDAPRID value identifies which product generated the accounting string.
  - If the requester is DB2, the accounting string is defined in QMDAINFO.
  - If QMDAPTP is DSN, QMDAINFO defines the format.
  - If QMDAPTP is SQL or JCC, QMDASQLI defines the format.
  - Otherwise, the format is undefined.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

Field Name: QMDAASTR
Locking

This topic shows detailed information about “Accounting - Locking”.

This block provides locking information. Locking ensures the integrity of data.

For formatting reasons, OMEGAMON XE for DB2 PE shows different labels for report and trace. The following example shows both layouts, the report on the left, and the trace layout on the right.

**Accounting - Locking**

The field labels shown in the following sample layout of “Accounting - Locking” are described in the following section.

<table>
<thead>
<tr>
<th>Report:</th>
<th>Trace:</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOCKING</td>
<td>LOCKING</td>
</tr>
<tr>
<td>AVERAGE</td>
<td>TOTAL</td>
</tr>
<tr>
<td>TIMEOUTS</td>
<td>0.00</td>
</tr>
<tr>
<td>DEADLOCKS</td>
<td>0.00</td>
</tr>
<tr>
<td>ESCAL.(SHARED)</td>
<td>0.00</td>
</tr>
<tr>
<td>ESCAL.(EXCLUS)</td>
<td>0.00</td>
</tr>
<tr>
<td>MAX PG/ROW LOCKS HELD</td>
<td>0.00</td>
</tr>
<tr>
<td>LOCK REQUEST</td>
<td>0.00</td>
</tr>
<tr>
<td>UNLOCK REQUEST</td>
<td>0.00</td>
</tr>
<tr>
<td>QUERY REQUEST</td>
<td>0.00</td>
</tr>
<tr>
<td>CHANGE REQUEST</td>
<td>0.00</td>
</tr>
<tr>
<td>OTHER REQUEST</td>
<td>0.00</td>
</tr>
<tr>
<td>TOTAL SUSPENSIONS</td>
<td>0.00</td>
</tr>
<tr>
<td>LOCK SUSPENSIONS</td>
<td>0.00</td>
</tr>
<tr>
<td>IRLM LATCH SUSPENS.</td>
<td>0.00</td>
</tr>
<tr>
<td>OTHER SUSPENS.</td>
<td>0.00</td>
</tr>
</tbody>
</table>

**TIMEOUTS**

The number of times a unit of work was suspended for a time exceeding the timeout value. This number should be low; ideally 0.

**Field Name:** QTXATIM

**DEADLOCKS**

The number of times deadlocks were detected. This number should be low, ideally 0.

**Background and Tuning Information**

Deadlocks occur when two or more application processes each hold locks on resources that the others need, without which they cannot proceed. Ensure that all applications accessing the same tables access them in the same order.

Deadlocks can also occur through index page splits if there is high insert activity. In this case, the recommendation is to set SUBPAGES to 1 for the index.

This field is incremented once for each deadlock encountered. There is no correlation between this field and the deadlock events reported in the Locking report set or the number of IFCID 172 records written. This field reports all deadlocks, regardless of how they were resolved. The locking report and record trace IFCID 172 show only those deadlocks that were resolved by DB2.

**Field Name:** QTXADEA
Locking

This is an exception field.

**ESCAL.(SHARED)/ESCAL.(SHAR)**

The number of times the maximum page locks per table space are exceeded, and the table space lock escalates from a page lock (IS) to a table space lock (S) for this thread. You can specify the number of locks allowed per table space with the LOCKS PER TABLE(SPACE) parameter on the DB2 install panel DSNTIPJ.

**Background and Tuning Information**

Escalations can cause unpredictable response times. Lock escalations should only happen when an application process updates or references (if repeatable read is used) more pages than normal.

**Field Name:** QTXALES

This is an exception field.

**ESCAL.(EXCLUS)/ESCAL.(EXCL)**

The number of times the maximum page locks per table space are exceeded and the table space lock escalates from a page lock (IX) to a table space lock (X).

**Background and Tuning Information**

Escalations can cause unpredictable response times. Lock escalations should only happen when an application process updates or references (if repeatable read is used) more pages than it normally does.

A useful rule of thumb is to compare the number of escalations (shared and exclusive) to the successful escalations (those that did not cause deadlocks and timeouts). If this value, or the number Lock escalations - shared and if the number of timeouts or deadlocks is also not 0, the timeout or deadlock is probably caused by the escalation.

If many escalations cause deadlocks and timeouts, the recommendation is to change the escalation threshold value. Use of ANY is extremely useful to prevent unnecessary and expensive page locks, for example locking all pages in a tablesce.

Lock escalations, shared or exclusive, should not be expected in a transaction environment.

**Field Name:** QTXALEX

This is an exception field.

**MAX PG/ROW LOCKS HELD/MAX PG/ROW LCK HELD**

The maximum number of page or row locks concurrently held against all table spaces by a single application during its execution. This count is a high-water mark. It cannot exceed the LOCKS PER USER parameter on panel DSNTIPJ.

**Field Name:** QTXANPL

This is an exception field.

**LOCK REQUEST**

The number of requests to lock a resource.

**Field Name:** QTXALOCK
Locking

This is an exception field.

UNLOCK REQUEST

The number of requests to unlock a resource.
This value can be less than the number of lock requests because DB2 can release several locks with a single unlock request.

Field Name: QTXAUNLK

QUERY REQUEST

The number of query requests.

Field Name: QTXAQRY

CHANGE REQUEST

The number of change requests.

Field Name: QTXACHG

OTHER REQUEST

The number of requests to IRLM to perform a function other than LOCK, UNLOCK, QUERY, or CHANGE.

Field Name: QTXAIRLM

TOTAL SUSPENSIONS

The number of all types of lock suspensions.

Field Name: ALRSUSP

LOCK SUSPENSIONS/LOCK SUSPENS

The number of times a lock could not be obtained and the unit of work was suspended.

Background and Tuning Information
This number should be low, ideally 0.
The number of lock suspensions is a function of the lock requests. Lock suspensions (or conflicts) can happen on either LOCK REQUEST or CHANGE REQUEST.
Suspensions are highly dependent on the application and table space locking protocols.

Field Name: QTXASLOC
This is an exception field.

IRLM LATCH SUSPENS.

The number of latch suspensions.
Field Name: QTXASLAT
This is an exception field.

OTHER SUSPENS.

The number of suspensions caused by something other than lock or latch.

Field Name: QTXASOTH
This is an exception field.
Logging Activity

This topic shows detailed information about “Accounting - Logging Activity”.

This block provides information about the logging activity.

The following example shows both layouts, the report on the left, and the trace layout on the right.

Accounting - Logging Activity

The field labels shown in the following sample layout of “Accounting - Logging Activity” are described in the following section.

<table>
<thead>
<tr>
<th>Report</th>
<th>Trace</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOGGING</td>
<td>LOGGING</td>
</tr>
<tr>
<td>AVERAGE</td>
<td>TOTAL</td>
</tr>
<tr>
<td>LOG RECORDS WRITTEN</td>
<td>2.00</td>
</tr>
<tr>
<td>TOT BYTES WRITTEN</td>
<td>6000.00</td>
</tr>
<tr>
<td>LOG RECORD SIZE</td>
<td>3000.00</td>
</tr>
</tbody>
</table>

LOG RECORDS WRITTEN (LOG RECS WRITTEN)

The number of log records written.

Field Name: QWAACLDRN

TOT BYTES WRITTEN (TOT BYTES WRITTEN)

The total number of log record bytes written.

Field Name: QWAACLDRAB

LOG RECORD SIZE

The average number of bytes written per log record.

Field Name: ALRAVGB
This topic shows detailed information about “Accounting - Miscellaneous”.

This block provides miscellaneous data for large objects (LOBs).

For formatting reasons, OMEGAMON XE for DB2 PE shows different labels for report and trace. The following example shows both layouts, the report on the left, and the trace layout on the right.

**Accounting - Miscellaneous**

The field labels shown in the following sample layout of “Accounting - Miscellaneous” are described in the following section.

<table>
<thead>
<tr>
<th>Report:</th>
<th>Trace:</th>
</tr>
</thead>
<tbody>
<tr>
<td>MISCELLANEOUS</td>
<td>MISCELLANEOUS</td>
</tr>
<tr>
<td>AVERAGE</td>
<td>TOTAL</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>----------------------------</td>
</tr>
<tr>
<td>MAX STO LOB VAL (KB)</td>
<td>0.00 0</td>
</tr>
<tr>
<td>MAX STO XML VAL (KB)</td>
<td>0.00 0</td>
</tr>
<tr>
<td>ARRAY EXPANSIONS</td>
<td>0.00 0</td>
</tr>
<tr>
<td>SPARSE IX DISABLED</td>
<td>0.00 0</td>
</tr>
<tr>
<td>SPARSE IX BUILT WF</td>
<td>0.00 0</td>
</tr>
<tr>
<td>MAX STO LOB VAL (KB)</td>
<td>MAX STO LOB VAL (KB)</td>
</tr>
<tr>
<td>MAX STO XML VAL (KB)</td>
<td>MAX STO XML VAL (KB)</td>
</tr>
<tr>
<td>ARRAY EXPANSIONS</td>
<td>ARRAY EXPANSIONS</td>
</tr>
<tr>
<td>SPARSE IX DISABLED</td>
<td>SPARSE IX DISABLED</td>
</tr>
<tr>
<td>SPARSE IX BUILT WF</td>
<td>SPARSE IX BUILT WF</td>
</tr>
</tbody>
</table>

**MAX STO LOB VAL (KB)**

Maximum storage used for LOB values.

*Field Name:* QXSTLOBV

**MAX STO XML VAL (KB)**

Maximum storage used for XML values.

*Field Name:* QXSTXMLV

**ARRAY EXPANSIONS**

The number of times a variable array has been expanded beyond 32 KB (DB2 field: QXSTARRAY_EXPANSIONS).

*Field Name:* AXSTAEXP

**SPARSE IX DISABLED**

The number of times that sparse index was disabled because of insufficient storage.

*Field Name:* QXSISTOR

**SPARSE IX BUILT WF**

The number of times that sparse-index built a physical work file for probing.

*Field Name:* QXSIWF
MVS Accounting

This topic shows detailed information about “Accounting - MVS Accounting”.

This block provides information about an MVS accounting trace.

Accounting - MVS Accounting

The field labels shown in the following sample layout of “Accounting - MVS Accounting” are described in the following section.

MVS ACCOUNTING DATA : IS-255-01234567890123456789012#2
ACCOUNTING TOKEN(CHAR) : N/A
ACCOUNTING TOKEN(HEX) : N/A

MVS ACCOUNTING DATA

The MVS accounting string associated with the MVS address space of the SQL application. It is filled if PROD_TYP=D; otherwise X’00’ is used.

This information comes from the ACCT= parameter on the job statement. If the ACCT= parameter is blank, the information on the EXEC statement is used. TSO logon Accounting information is used only if there is a value in the account field on the TSO Logon panel. Do not confuse this field with the Accounting correlation token.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

Field Name: QMDAACCT

ACCOUNTING TOKEN(CHAR)

The accounting token. For RRSAF, this is the RRSAF accounting token defined during signon. For DDF, this is the DDF correlation token.

This value is displayed in character format.

Field Name: AWHCTOKC

ACCOUNTING TOKEN(HEX)

The accounting token. For RRSAF, this is the RRSAF accounting token defined during signon. For DDF, this is the DDF correlation token.

This value is displayed in hexadecimal format.

Field Name: AWHCTOKH
Package Buffer Pool Activity - Class 10

This topic shows detailed information about “Accounting - Package Buffer Pool Activity - Class 10”.

This block shows buffer pool information at package level. It is repeated for each package present in the requested report. The block is headed by the package name.

The following example shows both layouts, the report on the left, and the trace layout on the right.

### Accounting - Package Buffer Pool Activity - Class 10

The field labels shown in the following sample layout of “Accounting - Package Buffer Pool Activity - Class 10” are described in the following section.

#### Report:

<table>
<thead>
<tr>
<th>NSQDLV</th>
<th>AVERAGE</th>
<th>TOTAL</th>
<th>DSNTP2</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>BPOOL HIT RATIO (%)</td>
<td>N/C</td>
<td>N/A</td>
<td>BPOOL HIT RATIO (%)</td>
<td>100</td>
</tr>
<tr>
<td>GETPAGES</td>
<td>0.00</td>
<td>0</td>
<td>GETPAGES</td>
<td>330201</td>
</tr>
<tr>
<td>BUFFER UPDATES</td>
<td>0.00</td>
<td>0</td>
<td>BUFFER UPDATES</td>
<td>0</td>
</tr>
<tr>
<td>SYNCHRONOUS WRITE</td>
<td>0.00</td>
<td>0</td>
<td>SYNCHRONOUS WRITE</td>
<td>0</td>
</tr>
<tr>
<td>SYNCHRONOUS READ</td>
<td>0.00</td>
<td>0</td>
<td>SYNCHRONOUS READ</td>
<td>0</td>
</tr>
<tr>
<td>SEQ. PREFETCH REQS</td>
<td>0.00</td>
<td>0</td>
<td>SEQ. PREFETCH REQS</td>
<td>0</td>
</tr>
<tr>
<td>LIST PREFETCH REQS</td>
<td>0.00</td>
<td>0</td>
<td>LIST PREFETCH REQS</td>
<td>0</td>
</tr>
<tr>
<td>DYN. PREFETCH REQS</td>
<td>0.00</td>
<td>0</td>
<td>DYN. PREFETCH REQS</td>
<td>0</td>
</tr>
<tr>
<td>PAGES READ ASYNCHR.</td>
<td>0.00</td>
<td>0</td>
<td>PAGES READ ASYNCHR.</td>
<td>0</td>
</tr>
</tbody>
</table>

#### BPOOL HIT RATIO (%)

The percentage of Getpage operations that were satisfied by a page already in the buffer pool.

The value is calculated as the ratio of number of successful Getpage operations minus the number of pages read from DASD (both synchronously and using prefetch), to the number of successful Getpage operations, expressed as a percentage.

#### Background and Tuning Information

The highest possible hit ratio is 100%, that is, when every page requested is always in the buffer pool. If the requested page is not in the buffer pool, the hit ratio is 0% or less. If the hit ratio is negative, this means that prefetch brought pages into the buffer pool that are not subsequently referenced, either because the query stops before it reaches the end of the table space, or because the prefetched pages are stolen by DB2 for reuse before the query can access them. A low buffer pool hit ratio is not necessarily bad. The hit ratio is a relative value, based on the type of application. For example, an application that browses large data might have a buffer pool hit ratio of 0. Watch for those cases where the hit ratio drops significantly for the same application. Here are some suggestions to increase the buffer hit ratio:

- Run the REORG utility for indexes or table spaces associated with the virtual buffer pool.
- Reserve more pages for random I/O by setting the SEQUENTIAL STEAL THRESHOLD (VPSEQT) to a lower value.
- Increase the buffer pool as long as the cost of paging does not outweigh the benefit of I/O avoidance.
Establish more separate buffer pools, perhaps to isolate different applications.

The hit ratio measurement becomes less meaningful if the buffer pool is used by additional processes, such as utilities or work files.

**Field Name:** ABUFFRAP

**GETPAGES**

The number of Getpage requests. This counter is incremented by successful Getpage requests for queries processed in parallel for each thread and for all successful and unsuccessful Getpage requests for queries that are not processed in parallel.

**Background and Tuning Information**

Reducing the number of Getpages can improve DB2 performance by reducing the number of synchronous page reads. With fewer Getpages, the requested page is more likely to be returned from the buffer pool. CPU usage is also reduced.

Check the ratio of Getpages to SQL DML statements, as a rule of thumb, try and keep this ratio below six.

You might need to modify the database and query design, for example:

- Add indexes to tables to reduce the number of pages scanned.
- Reassess the number of tables used and denormalize them, if necessary.
  - As an example, a large table with many columns can result in several pages being fetched to satisfy a simple query requesting just a few columns. Splitting such a table into several tables with fewer columns, tailored to queries, will result in fewer pages returned for each query.
- Use correlated rather than noncorrelated queries to force the use of an index.

**Field Name:** QBACGETP

**BUFFER UPDATES**

The number of times a buffer update occurs. This is incremented every time a page is updated and is ready to be written to DASD. If the same page is updated twice, for example, the number is incremented by 2.

This number is kept for all types of pages including data pages and work-file pages.

**Background and Tuning Information**

A nonzero value indicates any of the following activities:

- SQL INSERT, UPDATE, or DELETE
- Merge scan join
- Internal sort activity on the work files

Check the access path to determine whether sort activity can be minimized or avoided.

**Field Name:** QBACSWSP

**SYNCHRONOUS WRITE**

The number of immediate (synchronous) write I/O operations.

**Background and Tuning Information**
Although an immediate write is rare, a small nonzero value is acceptable. A large value indicates that the system needs tuning.

Field Name: QBACIMWP

SYNCHRONOUS READ

The number of synchronous read I/O operations. DB2 increments this counter for each media manager synchronous physical read. Asynchronous I/O requests are not counted.

Field Name: QBACRIOP

SEQ. PREFETCH REQS

The number of SEQUENTIAL PREFETCH requests. This is incremented for each PREFETCH request. Each request can result in an I/O read. If it does, up to 32 pages can be read for SQL and up to 64 pages for utilities. For SQL, depending on the buffer pool size, a request does not result in an I/O if all the requested pages are already in the buffer pool.

DB2 can use sequential prefetch if the data is accessed in sequential order even though sequential prefetch was not requested at bind time. This is known as sequential detection and is not included in the sequential prefetch count. Sequential detection is included in dynamic prefetch requests field.

Background and Tuning Information

Table space scans and nonmatching index scans generally use sequential prefetch.

Field Name: QBACSEQP

LIST PREFETCH REQS

The number of LIST PREFETCH requests.

Special Considerations:
1. List prefetch allows DB2 to access data pages efficiently even if the needed data pages are not contiguous. It can be used with single index access and is always used with multiple index access.
2. List prefetch is always used to access data from the inner table during a hybrid join.
3. Data pages are read in quantities equal to the sequential prefetch quantity, which depends on the buffer pool size and is usually 32 pages.
4. During bind time DB2 does not use list prefetch if the estimated number of RIDs to be processed would take more than 50% of the RID pool. During execution time, list prefetch processing terminates if DB2 detects that more than 25% of the rows in the table need to be accessed. If list prefetch is terminated, it is indicated in IFCID 125.

Field Name: QBACLPFP

DYN. PREFETCH REQS

The number of (dynamic) PREFETCH requests. This is triggered by sequential detection. This includes prefetches for segmented table spaces.

Background and Tuning Information

Dynamic prefetch is typically used for a SELECT or UPDATE that is run repeatedly, accessing the index for each access.
If sequential prefetch, list prefetch, and dynamic prefetch reads have large values, check whether the access path can be improved.

Field Name: QBACDPFP

PAGES READ ASYNCHR.

The number of asynchronous pages read by prefetch that the agent triggered.

Background and Tuning Information

This is used to determine the buffer pool hit ratio: (Getpage requests - Synchronous reads - Asynchronous pages read) / Getpage requests.

Field Name: QBACSIOP
Package Global Contention L-Locks - Class 8

Package Global Contention L-Locks - Class 8

This topic shows detailed information about “Accounting - Package Global Contention L-Locks - Class 8.”

This block provides global contention information for a logical lock (L-lock) at package level. It shows conflicts on locking requests between different DB2 members of a data sharing group when those members are trying to serialize shared resources.

Note: The current package name is shown in the header line of the block instead of this block title.

Accounting - Package Global Contention L-Locks - Class 8

The field labels shown in the following sample layout of “Accounting - Package Global Contention L-Locks - Class 8” are described in the following section.

<table>
<thead>
<tr>
<th>Report:</th>
<th>Trace:</th>
</tr>
</thead>
<tbody>
<tr>
<td>AYRSSDO20</td>
<td>AVERAGE TIME</td>
</tr>
<tr>
<td>GLOBAL CONTENTION L-LOCKS</td>
<td>0.000000</td>
</tr>
<tr>
<td>PARENT (DB,TS,TAB,PART)</td>
<td>0.000000</td>
</tr>
<tr>
<td>CHILD (PAGE,ROW)</td>
<td>0.000000</td>
</tr>
<tr>
<td>OTHER</td>
<td>0.000000</td>
</tr>
</tbody>
</table>

GLOBAL CONTENTION L-LOCKS - AVERAGE TIME/ELAPSED TIME

The accumulated wait times due to global contention for all L-Locks.

Field Name: APLKSUST

GLOBAL CONTENTION L-LOCKS - AV.EVENT/EVENTS

The accumulated wait trace events processed for waits for global contention of all L-Locks.

Field Name: APLKSUSC

PARENT (DB,TS,TAB,PART) - AVERAGE TIME/ELAPSED TIME

The accumulated wait time due to global contention for parent L-Locks. Parent L-Locks are any of the following L-Lock types: database, tablespace, table, or partition.

Field Name: QPACAWTJ

PARENT (DB,TS,TAB,PART) - AV.EVENT/EVENTS

The number of wait trace events processed for waits for global contention for parent L-Locks.

Field Name: APLPSUSC

CHILD (PAGE,ROW) - AVERAGE TIME/ELAPSED TIME

The accumulated wait time due to global contention for child L-Locks. Child L-locks are any of the following L-Lock types: page or row.

Field Name: QPACAWTK

CHILD (PAGE,ROW) - AV.EVENT/EVENTS

The number of wait trace events processed for waits for global contention for child L-Locks.

Field Name: APLCSUSC

OTHER - AVERAGE TIME/ELAPSED TIME
Package Global Contention L-Locks - Class 8

The accumulated wait time due to global contention for other L-Locks.

Field Name: QPACAWTM

OTHER - AV.EVENT/EVENTS

The number of wait trace events processed for waits for global contention for other L-Locks.

Field Name: APLOUSC
Package Global Contention P-Locks - Class 8

Package Global Contention P-Locks - Class 8

This topic shows detailed information about “Accounting - Package Global Contention P-Locks - Class 8”.

This block provides global contention information for a physical lock (P-lock) at package level. It shows conflicts on locking requests between different DB2 members of a data sharing group when those members are trying to serialize shared resources.

The following example shows both layouts, the report on the left, and the trace layout on the right.

Note: The current package name is shown in the header line of the block instead of this block title.

Accounting - Package Global Contention P-Locks - Class 8

The field labels shown in the following sample layout of “Accounting - Package Global Contention P-Locks - Class 8” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPLSUST</td>
<td>GLOBAL CONTENTION P-LOCKS - AVERAGE TIME/ELAPSED TIME</td>
</tr>
<tr>
<td>APPLSUSC</td>
<td>GLOBAL CONTENTION P-LOCKS - AV.EVENT/EVENTS</td>
</tr>
<tr>
<td>QPACAWTN</td>
<td>PAGESET/PARTITION - AVERAGE TIME/ELAPSED TIME</td>
</tr>
<tr>
<td>APPSSUSC</td>
<td>PAGE - AVERAGE TIME/ELAPSED TIME</td>
</tr>
<tr>
<td>APPPSUSC</td>
<td>PAGE - AV.EVENT/EVENTS</td>
</tr>
</tbody>
</table>

GLOBAL CONTENTION P-LOCKS - AVERAGE TIME/ELAPSED TIME

The accumulated wait times due to global contention for all P-Locks.

Field Name: APPLSUST

GLOBAL CONTENTION P-LOCKS - AV.EVENT/EVENTS

The accumulated wait trace events processed for waits for global contention of all P-Locks.

Field Name: APPLSUSC

PAGESET/PARTITION - AVERAGE TIME/ELAPSED TIME

The accumulated wait time due to global contention for pageset/partition P-Locks.

Field Name: QPACAWTN

PAGESET/PARTITION - AV.EVENT/EVENTS

The number of wait trace events processed for waits for global contention for pageset/partition P-Locks.

Field Name: APPSSUSC

PAGE - AVERAGE TIME/ELAPSED TIME

The accumulated wait time due to global contention for page P-Locks.

Field Name: QPACAWTO

PAGE - AV.EVENT/EVENTS

The number of wait trace events processed for waits for global contention for page P-Locks.

Field Name: APPPSUSC
Package Global Contention P-Locks - Class 8

OTHER - AVERAGE TIME/ELAPSED TIME
The accumulated wait time due to global contention for other P-Locks.
Field Name: QPACAWTQ

OTHER - AV.EVENT/EVENTS
The number of wait trace events processed for waits for global contention for other P-Locks.
Field Name: APPOSUSC
Package Identification

Package Identification

This topic shows the report and trace blocks for package identification. They present information for the identification of packages. A package is an object containing a set of SQL statements that have been statically bound and that is available for processing.

For more information on calculating package average data refer to “How Averages Are Calculated” on page 5-4.

“Package Identification - Report” on page 5-137
This topic shows detailed information about “Accounting - Package Identification - Report”.

“Package Identification - Trace” on page 5-142
This topic shows detailed information about “Accounting - Package Identification - Trace”.
## Package Identification - Report

### Accounting - Package Identification - Report

The field labels shown in the following sample layout of “Accounting - Package Identification - Report” are described in the following section.

**ACTNAME:** STORED PROC.CREATETABLE

<table>
<thead>
<tr>
<th>CREATETA</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>TYPE</td>
<td>PACKAGE</td>
</tr>
<tr>
<td>LOCATION</td>
<td>DSNDA1B</td>
</tr>
<tr>
<td>COLLECTION ID</td>
<td>USRT001</td>
</tr>
<tr>
<td>PROGRAM NAME</td>
<td>NSQLDLV</td>
</tr>
</tbody>
</table>

**ACTIVITY TYPE** NATIVE SQL PROC

**ACTIVITY NAME** NSQLDLV

**SCHEMA NAME** USRT001

**SUCC AUTH CHECK** 0

**OCCURRENCES** 478960

**NBR OF ALLOCATIONS** 28986

**SQL STMT - AVERAGE** N/P

**SQL STMT - TOTAL** N/P

**NBR RLUP THREADS** 28985

**PCKNAME**

This label is replaced by the package name, or, if ORDER (ACTNAME) was in effect, the package activity name. An activity name is truncated if it is longer than 8 characters.

**Field Name:** PCKNAME

**TYPE**

An indicator of whether the block describes a package or a DBRM. Possible values are PACKAGE, DBRM, and BOTH. BOTH can be shown in reports if there are packages and DBRMs with the same program name.

**Field Name:** ADPCKTYP

**LOCATION**

The location name.

If this field is blank in trace or report, the package or DBRM was executed locally. If it is not blank, all times represent the time spent locally to execute the remote package for this APPL_DIR requester.

This field is invalid (N/P) in case of DB2 10 or later if summary rollup data is present.

**Field Name:** QPACLOCN

This is an exception field.

**COLLECTION ID**

The package collection ID. This field does not apply to DBRMs. If the program name cannot be identified, this field is not present in report or trace.
This field is invalid in case of DB2 10 or later if summary rollup data is present. It can have the following value in:

- Accounting trace and report: N/P
- The Accounting FILE and SAVE PROGRAM table: blank

**Field Name:** QPACCOLN

This is an *exception* field.

### PROGRAM NAME

The program name (package ID or DBRM name).

In the case of rollup data (Accounting data of DDF/RRSAF threads and parallel tasks accumulated by DB2), the following value is shown:

- *ROLSUM* for DB2 10 or later
- *ROLLUP* for DB2 versions prior to DB2 10

**Field Name:** QPACPKID

This is an *exception* field.

### PROGRAM NAME

This field is identical to QPACPKID except of when ORDER (ACTNAME) was in effect. can belong to the same activity name. In a data block that reports totals it is set to ALL PROG.

**Field Name:** APACPKID

### ACTIVITY TYPE

The type of activity. The following values indicate how the package was loaded:

- **ALL TYPES**
  
  In a data block that reports totals it is set to ALL TYPES.

- **STORED PROC**
  
  When running an external procedure

- **TRIGGER**
  
  When running a trigger

- **UDF**
  
  When running a user-defined function

- **NATIVE SQL PROC**
  
  When running a native SQL procedure

- **NATIVE UDF**
  
  When running a native UDF procedure (a non-inline user-defined function)

- **NONNESTED**
  
  Indicates that none of the above values is true

- **MULTIPLE**
  
  Indicates that packages with the same key but with different activity types were running

- **N/P**
  
  Invalidated in case of rollup summary

The nested activity values that are shown in column NEST_ACTIVITY_TYPE of the table DB2PMFACCT_PROGRAM are:

- **S**
  
  For Stored Procedure
Package Identification - Report

| T | For Trigger |
| U | For UDF |
| Q | For native SQL procedure |
| D | For Native UDF |
| N | For nonnested (other) |
| blank | For invalidated in case of rollup summary |

This field is invalid in case of DB2 10 or later if unique or summary rollup data is present.

Field Name: ADPATYP

ACTIVITY NAME

The name of the nested activity.

This field contains the name of the nested activity if the package is defined for:

- Trigger
- Stored procedure
- User-defined function (UDF)
- Native SQL procedure
- Non-inline UDF

In a data block that reports totals it is set to ALL NAMES.

This field is invalid in case of DB2 10 or later if summary rollup data is present.

It can have the following value in:

- Accounting Trace and Report: N/P
- The Accounting FILE and SAVE PROGRAM tables: blank

Field Name: ADPAANM

SCHEMA NAME

Schema name of the nested activity.

If the package is defined for a trigger, stored procedure, or user-defined function, then this field contains the name of the schema to which the nested activity belongs. It can have the following value in:

- Accounting Trace and Report: N/P
- The Accounting FILE and SAVE PROGRAM tables: blank

This field is invalid in case of DB2 10 or later if summary rollup data is present.

Field Name: ADPASCH

SUCC AUTH CHECK

For Accounting reports, this field shows the number of times authorization information was found for this package without accessing the DB2 catalog.

For DB2 10 or later a value of 0 is shown. This field is valid for non-rollup data.

Field Name: ADPCKANR

OCCURRENCES
This value can be one of the following:

- In general, the total number of accounting trace sections for a package or DBRM regardless of enabled or disabled DB2 trace classes 7 and 8 at the time of writing the trace record. In case of Distributed Data Facility (DDF) or Recoverable Resource Manager Services Attach Facility (RRSAF) threads, it is the number of accounting intervals rolled up in a record.

- If ORDER (ACTNAME) is specified, the total number of package sections of a special activity type depends on the following:
  - If IFCIDs 233 are available, stored procedures (SP) and user-defined functions (UDF) themselves are counted based on IFCID 233. Subprograms called by these routines and functions are not taken into account.
  - If IFCIDs 233 are not collected, all packages of an activity type are counted. The sum also includes the number of subprograms.

Field Name: ADTOTPOC

NBR OF ALLOCATIONS

This value can be one of the following:

- In general, the number of times a package was invoked by a different package. For the first package run by an application, the initial call counts as a package switch. If this package called a nested package (such as a trigger, UDF, or stored procedure), a switch will not be counted upon return from such a package.

- If ORDER (ACTNAME) is specified, the number of times a package of a special activity type is invoked from a different package depends on the following:
  - If IFCIDs 233 are available, the invocations of stored procedures (SP) and user-defined functions (UDF) themselves are counted based on IFCID 233. Invocations of subprograms called by these routines and functions are not taken into account.
  - If IFCIDs 233 are not collected, all invocations of an activity type are counted. The sum also includes the number of subprograms.

Field Name: APACSWIT

SQL STMT - AVERAGE

The number of SQL statements issued in this package or DBRM.

This number may not be equal to the total number of SQL statements in the QXST data section because QXST does not count all SQL statements. For example, it does not count commit or rollback statements.

Note: This field is shown for the following field labels in Accounting trace:

- SQL STMT - TOTAL
- SQL STMT - AVERAGE:
  - For DB2 9, the average is not calculated because it is identical to the TOTAL value. N/C (not calculated) is shown for this field.
  - For DB2 10 or later, the average data is shown.

Field Name: QPACSQLC

This is an exception field.
SQL STMT - TOTAL
The number of SQL statements issued in this package or DBRM.
This number may not be equal to the total number of SQL statements in
the QXST data section because QXST does not count all SQL statements.
For example, it does not count commit or rollback statements.

Note: This field is shown for the following field labels in Accounting
trace:
- SQL STMT - TOTAL
- SQL STMT - AVERAGE:
  - For DB2 9, the average is not calculated because it is identical to the
    TOTAL value. N/C (not calculated) is shown for this field.
  - For DB2 10 or later, the average data is shown.

Field Name: QPACSQLC
This is an exception field.

NBR RLUP THREADS
This value can be one of the following:
- In general, the number of threads to roll data into this QPAC data
  section. Non-rollup QPACs have a value of 1 and rollup QPACs have a
  value of 1 or more. With DB2 10 or later, this number is used as a
divisor for calculating averages for package class 7, 8, or 10 times and
events.
- If ORDER (ACTNAME) is specified, the number of threads to roll data
  into this QPAC data section of a special activity type depends on the
following:
  - If IFCIDs 233 are available, the number of threads to roll data into
    this QPAC data section for a stored procedure (SP) or user-defined
    function (UDF). That number of subprograms called by these routines
    and functions is not taken into account.
  - If IFCIDs 233 are not collected, the total number of threads to roll
data into this QPAC data section. The sum also includes the number
of subprograms.

Field Name: QPACRLNU
Package Identification - Trace

Package Identification - Trace
This topic shows detailed information about “Accounting - Package Identification - Trace”.

Accounting - Package Identification - Trace
The field labels shown in the following sample layout of “Accounting - Package Identification - Trace” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCKNAME</td>
<td>This label is replaced by the package name, or, if ORDER (ACTNAME) was in effect, the package activity name. An activity name is truncated if it is longer than 8 characters.</td>
</tr>
<tr>
<td>TYPE</td>
<td>An indicator of whether the block describes a package or a DBRM. Possible values are PACKAGE, DBRM, and BOTH. BOTH can be shown in reports if there are packages and DBRMs with the same program name.</td>
</tr>
<tr>
<td>LOCATION</td>
<td>The location name. If this field is blank in trace or report, the package or DBRM was executed locally. If it is not blank, all times represent the time spent locally to execute the remote package for this APPL_DIR requester. This field is invalid (N/P) in case of DB2 10 or later if summary rollup data is present.</td>
</tr>
<tr>
<td>COLLECTION ID</td>
<td>The package collection ID. This field does not apply to DBRMs. If the program name cannot be identified, this field is not present in report or trace. This field is invalid in case of DB2 10 or later if summary rollup data is present. It can have the following value in:</td>
</tr>
<tr>
<td></td>
<td>• Accounting trace and report: N/P</td>
</tr>
<tr>
<td></td>
<td>• The Accounting FILE and SAVE PROGRAM table: blank</td>
</tr>
</tbody>
</table>
Field Name: QPACCOLN
This is an exception field.

PROGRAM NAME
The program name (package ID or DBRM name).
In the case of rollup data (Accounting data of DDF/RRSAF threads and parallel tasks accumulated by DB2), the following value is shown:
• *ROLSUM* for DB2 10 or later
• *ROLLUP* for DB2 versions prior to DB2 10

Field Name: QPACPKID
This is an exception field.

CONSISTENCY TOKEN
The program (package or DBRM) consistency token.
This field is invalid (0) in case of DB2 10 or later if summary rollup data is present.

Field Name: QPACCONT

ACTIVITY TYPE
The type of activity. The following values indicate how the package was loaded:

ALL TYPES
In a data block that reports totals it is set to ALL TYPES.
STORED PROC
When running an external procedure
TRIGGER
When running a trigger
UDF
When running a user-defined function
NATIVE SQL PROC
When running a native SQL procedure
NATIVE UDF
When running a native UDF procedure (a non-inline user-defined function)
NONNESTED
Indicates that none of the above values is true
MULTIPLE
Indicates that packages with the same key but with different activity types were running
N/P
Invalidated in case of rollup summary

The nested activity values that are shown in column NEST_ACTIVITY_TYPE of the table DB2PMFACCT_PROGRAM are:
S
For Stored Procedure
T
For Trigger
U
For UDF
Q
For native SQL procedure
Package Identification - Trace

D For Native UDF
N For nonnested (other)
blank For invalidated in case of rollup summary

This field is invalid in case of DB2 10 or later if unique or summary rollup data is present.

Field Name: ADPATYP

ACTIVITY NAME

The name of the nested activity.

This field contains the name of the nested activity if the package is defined for a:

- Trigger
- Stored procedure
- User-defined function (UDF)
- Native SQL procedure
- Non-inline UDF

In a data block that reports totals it is set to ALL NAMES.

This field is invalid in case of DB2 10 or later if summary rollup data is present.

It can have the following value in:

- Accounting Trace and Report: N/P
- The Accounting FILE and SAVE PROGRAM tables: blank

Field Name: ADPAANM

SCHEMA NAME

Schema name of the nested activity.

If the package is defined for a trigger, stored procedure, or user-defined function, then this field contains the name of the schema to which the nested activity belongs. It can have the following value in:

- Accounting Trace and Report: N/P
- The Accounting FILE and SAVE PROGRAM tables: blank

This field is invalid in case of DB2 10 or later if summary rollup data is present.

Field Name: ADPASCH

SUCC AUTH CHECK

For Accounting traces, this field indicates whether authorization information was found for this package without accessing the DB2 catalog. This field is valid for non-rollup data. Possible values are:

- YES
- NO
- N/A if DB2 or later is used
- N/P, this field is invalid for Accounting trace
- blank, this field is invalid in the Accounting FILE PROGRAM table
Note: This field is invalid in case of DB2 10 or later if unique or summary rollup data is present.

**Field Name:** ADPCKAUT

**NBR OF ALLOCATIONS**

This value can be one of the following:

- In general, the number of times a package was invoked by a different package. For the first package run by an application, the initial call counts as a package switch. If this package called a nested package (such as a trigger, UDF, or stored procedure), a switch will **not** be counted upon return from such a package.

- If ORDER (ACTNAME) is specified, the number of times a package of a special activity type is invoked from a different package depends on the following:
  - If IFCIDs 233 are available, the invocations of stored procedures (SP) and user-defined functions (UDF) themselves are counted based on IFCID 233. Invocations of subprograms called by these routines and functions are not taken into account.
  - If IFCIDs 233 are not collected, all invocations of an activity type are counted. The sum also includes the number of subprograms.

**Field Name:** APACSWIT

**SQL STMT - AVG**

The number of SQL statements issued in this package or DBRM. This number may not be equal to the total number of SQL statements in the QXST data section because QXST does not count all SQL statements. For example, it does not count commit or rollback statements.

**Note:** This field is shown for the following field labels in Accounting trace:

- SQL STMT - TOTAL
- SQL STMT - AVERAGE:
  - For DB2 9, the average is not calculated because it is identical to the TOTAL value. N/C (not calculated) is shown for this field.
  - For DB2 10 or later, the average data is shown.

**Field Name:** QPACSQLC

This is an *exception* field.

**SQL STMT - TOTAL**

The number of SQL statements issued in this package or DBRM. This number may not be equal to the total number of SQL statements in the QXST data section because QXST does not count all SQL statements. For example, it does not count commit or rollback statements.

**Note:** This field is shown for the following field labels in Accounting trace:

- SQL STMT - TOTAL
- SQL STMT - AVERAGE:
  - For DB2 9, the average is not calculated because it is identical to the TOTAL value. N/C (not calculated) is shown for this field.
For DB2 10 the average data is shown.

Field Name: QPACSQLC

This is an *exception* field.

**NBR RLUP THREADS**

This value can be one of the following:

- In general, the number of threads to roll data into this QPAC data section. Non-rollup QPACs have a value of 1 and rollup QPACs have a value of 1 or more. With DB2 10 or later, this number is used as a divisor for calculating averages for package class 7, 8, or 10 times and events.

- If ORDER (ACTNAME) is specified, the number of threads to roll data into this QPAC data section of a special activity type depends on the following:
  - If IFCIDs 233 are available, the number of threads to roll data into this QPAC data section for a stored procedure (SP) or user-defined function (UDF). That number of subprograms called by these routines and functions is not taken into account.
  - If IFCIDs 233 are not collected, the total number of threads to roll data into this QPAC data section. The sum also includes the number of subprograms.

Field Name: QPACRLNU
Package Locking Activity - Class 10

This topic shows detailed information about “Accounting - Package Locking Activity - Class 10”.

This block shows locking information at package level. It is repeated for each package present in the requested report. The block is headed by the package name.

The following example shows both layouts, the report on the left, and the trace layout on the right.

**Accounting - Package Locking Activity - Class 10**

The field labels shown in the following sample layout of “Accounting - Package Locking Activity - Class 10” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TIMEOUTS</td>
<td>The number of times a unit of work was suspended for a time exceeding the timeout value. This number should be low, ideally 0.</td>
</tr>
<tr>
<td>DEADLOCKS</td>
<td>The number of times deadlocks were detected. This number should be low, ideally 0.</td>
</tr>
</tbody>
</table>

**Background and Tuning Information**

Deadlocks occur when two or more application processes each hold locks on resources that the others need, without which they cannot proceed. Ensure that all applications accessing the same tables access them in the same order.

Deadlocks can also occur through index page splits if there is high insert activity. In this case, the recommendation is to set SUBPAGES to 1 for the index.

This field is incremented once for each deadlock encountered. There is no correlation between this field and the deadlock events reported in the Locking report set or the number of IFCID 172 records written. This field reports all deadlocks, regardless of how they were resolved. The locking report and record trace IFCID 172 show only those deadlocks that were resolved by DB2.
Field Name: QTXADEA
This is an exception field.

ESCAL.(SHARED)
The number of times the maximum page locks per table space are exceeded, and the table space lock escalates from a page lock (IS) to a table space lock (S) for this thread. You can specify the number of locks allowed per table space with the LOCKS PER TABLE(SPACE) parameter on the DB2 install panel DSNTIPJ.

Background and Tuning Information
Escalations can cause unpredictable response times. Lock escalations should only happen when an application process updates or references (if repeatable read is used) more pages than normal.

Field Name: QTXALES
This is an exception field.

ESCAL.(EXCLUS)
The number of times the maximum page locks per table space are exceeded and the table space lock escalates from a page lock (IX) to a table space lock (X).

Background and Tuning Information
Escalations can cause unpredictable response times. Lock escalations should only happen when an application process updates or references (if repeatable read is used) more pages than it normally does.

A useful rule of thumb is to compare the number of escalations (shared and exclusive) to the successful escalations (those that did not cause deadlocks and timeouts). If this value, or the number Lock escalations - shared and if the number of timeouts or deadlocks is also not 0, the timeout or deadlock is probably caused by the escalation.

If many escalations cause deadlocks and timeouts, the recommendation is to change the escalation threshold value. Use of ANY is extremely useful to prevent unnecessary and expensive page locks, for example locking all pages in a tablespace.

Lock escalations, shared or exclusive, should not be expected in a transaction environment.

Field Name: QTXALEX
This is an exception field.

MAX PG/ROW LOCKS HELD
The maximum number of page or row locks concurrently held against all table spaces by a single application during its execution. This count is a high-water mark. It cannot exceed the LOCKS PER USER parameter on panel DSNTIPJ.

Field Name: QTXANPL
This is an exception field.

LOCK REQUEST
The number of requests to lock a resource.
Field Name: QTXALOCK
This is an exception field.

UNLOCK REQUEST
The number of requests to unlock a resource.
This value can be less than the number of lock requests because DB2 can release several locks with a single unlock request.
Field Name: QTXAUNLK

QUERY REQUEST
The number of query requests.
Field Name: QTXAQRY

CHANGE REQUEST
The number of change requests.
Field Name: QTXACHG

OTHER REQUEST
The number of requests to IRLM to perform a function other than LOCK, UNLOCK, QUERY, or CHANGE.
Field Name: QTXAIRLM

LOCK SUSPENSIONS
The number of times a lock could not be obtained and the unit of work was suspended.

Background and Tuning Information
This number should be low, ideally 0.
The number of lock suspensions is a function of the lock requests. Lock suspensions (or conflicts) can happen on either LOCK REQUEST or CHANGE REQUEST.
Suspensions are highly dependent on the application and table space locking protocols.
Field Name: QTXASLOC
This is an exception field.

IRLM LOCK SUSPENS.
The number of latch suspensions.
Field Name: QTXASLAT
This is an exception field.

OTHER SUSPENS.
The number of suspensions caused by something other than lock or latch.
Field Name: QTXASOTH
This is an exception field.
Package SQL Activity - Class 10

Package SQL Activity - Class 10

This topic shows detailed information about “Accounting - Package SQL Activity - Class 10”.

This block shows SQL information at package level. It is repeated for each package present in the requested report. The block is headed by the package name.

The following example shows both layouts, the report on the left, and the trace layout on the right.

Accounting - Package SQL Activity - Class 10

The field labels shown in the following sample layout of “Accounting - Package SQL Activity - Class 10” are described in the following section.

<table>
<thead>
<tr>
<th>Report:</th>
<th>Trace:</th>
</tr>
</thead>
<tbody>
<tr>
<td>package</td>
<td>AVERAGE</td>
</tr>
<tr>
<td>SELECT</td>
<td>0.00</td>
</tr>
<tr>
<td>INSERT</td>
<td>0.00</td>
</tr>
<tr>
<td>UPDATE</td>
<td>0.00</td>
</tr>
<tr>
<td>DELETE</td>
<td>0.00</td>
</tr>
<tr>
<td>DESCRIBE</td>
<td>0.00</td>
</tr>
<tr>
<td>PREPARE</td>
<td>0.00</td>
</tr>
<tr>
<td>OPEN</td>
<td>0.00</td>
</tr>
<tr>
<td>FETCH</td>
<td>0.00</td>
</tr>
<tr>
<td>CLOSE</td>
<td>0.00</td>
</tr>
<tr>
<td>LOCK TABLE</td>
<td>0.00</td>
</tr>
<tr>
<td>CALL</td>
<td>0.00</td>
</tr>
</tbody>
</table>

SELECT

The number of SQL SELECT statements executed.

Field Name: QPSELECT

This is an exception field.

INSERT

The number of INSERT statements executed.

Field Name: QPINSRT

This is an exception field.

UPDATE

The number of UPDATE statements executed.

Field Name: QPUPDTE

This is an exception field.

DELETE

The number of DELETE statements executed.

Field Name: QPDELET

This is an exception field.

DESCRIBE

The number of data capture describes.
Package SQL Activity - Class 10

Field Name: QPDESC

PREPARE
The number of full prepare requests.
Field Name: QPPREP

OPEN
The number of full open requests.
Field Name: QPOPEN

FETCH
The number of fetch requests.
Field Name: QPFETCH

CLOSE
The number of close requests.
Field Name: QPCLOSE

LOCK TABLE
The number of lock tables.
Field Name: QPLOCK

CALL
The number of SQL calls.
Field Name: QPCALL
Package Times - Class 8 - Suspensions

This topic shows detailed information about “Accounting - Package Times - Class 8 - Suspensions”.

This block provides suspension information for class 8.

The following example shows both layouts, the report on the left, and the trace layout on the right.

**Accounting - Package Times - Class 8 - Suspensions**

The field labels shown in the following sample layout of “Accounting - Package Times - Class 8 - Suspensions” are described in the following section.

<table>
<thead>
<tr>
<th>Report:</th>
<th>Trace:</th>
</tr>
</thead>
<tbody>
<tr>
<td>PACKAGE</td>
<td>AVERAGE TIME</td>
</tr>
<tr>
<td>LOCK/LATCH</td>
<td>0.000000</td>
</tr>
<tr>
<td>IRLM LOCK+LATCH</td>
<td>0.000000</td>
</tr>
<tr>
<td>DB2 LATCH</td>
<td>0.000000</td>
</tr>
<tr>
<td>SYNCHRONOUS I/O</td>
<td>0.000000</td>
</tr>
<tr>
<td>OTHER READ I/O</td>
<td>0.000000</td>
</tr>
<tr>
<td>OTHER WRITE I/O</td>
<td>0.000000</td>
</tr>
<tr>
<td>SERV.TASK SWITCH</td>
<td>0.000000</td>
</tr>
<tr>
<td>ARCH.LOB (QUIESCE)</td>
<td>0.000000</td>
</tr>
<tr>
<td>ARCHIVE LOG READ</td>
<td>0.000000</td>
</tr>
<tr>
<td>DRAIN LOCK</td>
<td>0.000000</td>
</tr>
<tr>
<td>CLAIM RELEASE</td>
<td>0.000000</td>
</tr>
<tr>
<td>PAGE LATCH</td>
<td>0.000000</td>
</tr>
<tr>
<td>NOTIFY MESSAGES</td>
<td>0.000000</td>
</tr>
<tr>
<td>GLOBAL CONTENTION</td>
<td>0.000000</td>
</tr>
<tr>
<td>TCP/IP LOB XML</td>
<td>0.000000</td>
</tr>
<tr>
<td>ACCELERATOR</td>
<td>0.000000</td>
</tr>
<tr>
<td>PQ SYNCHRONIZATION</td>
<td>0.000000</td>
</tr>
<tr>
<td>TOTAL CL8 SUSPENS.</td>
<td>0.000000</td>
</tr>
</tbody>
</table>

**PACKAGE**

This label is replaced by the package name, or, if ORDER (ACTNAME) was in effect, the package activity name. An activity name is truncated if it is longer than 8 characters.

**Field Name:** PACKAGE

**AVERAGE TIME - LOCK/LATCH**

This field depends on the DB2 version that is installed:

- **DB2 10 or later:** The sum of the number of wait trace events processed for waits for lock and the number of wait trace events processed for page latch contention while executing this package.

- **Prior to DB2 10:** The accumulated lock elapsed wait time that occurred while executing this package.

**Note:** The internally defined field adjusts the original DB2 value. DB2 counts each event twice, one for the entry and one for the exit.

**Field Name:** AWTPLOLA

**AVERAGE TIME - IRLM LOCK+LATCH**

This field depends on the DB2 version that is installed:

- **DB2 10 or later:** The accumulated lock elapsed wait time that occurred while executing this package.

- **Prior to DB2 10:** The accumulated elapsed time spent by the package or DBRM waiting for lock and latch suspensions.

**Background and Tuning Information**
Package Times - Class 8 - Suspensions

OMEGAMON XE for DB2 PE might adjust this value if the thread was suspended when its performance data was gathered.

If the suspension time is high, investigate locking activity.

**Field Name:** QPACAWTL

This is an *exception* field.

**AVERAGE TIME - DB2 LATCH**

The accumulated latch elapsed wait time for latch suspensions that occurred while executing this package.

**Field Name:** QPACAWLH

**AVERAGE TIME - SYNCHRONOUS I/O**

The accumulated elapsed wait time for I/O suspensions under this thread during the execution of the package or DBRM.

**Background and Tuning Information**

OMEGAMON XE for DB2 PE might adjust this value if the thread was suspended when its performance data was gathered.

**Field Name:** QPACAWTI

This is an *exception* field.

**AVERAGE TIME - OTHER READ I/O**

The accumulated waiting time for a read I/O performed under a thread other than this one during the execution of the package or DBRM.

**Background and Tuning Information**

OMEGAMON XE for DB2 PE might adjust this value if the thread was suspended when performance data was gathered.

This field includes waits caused by sequential prefetch, list prefetch, dynamic prefetch, and synchronous read I/O performed by other threads.

If the value in this field is high, the problem could be an I/O bound query using prefetch or an I/O contention. The application is accessing data from a busy data set, volume, or control unit and is continually being suspended. Consult the DBA and MVS systems programmer.

**Field Name:** QPACAWTR

This is an *exception* field.

**AVERAGE TIME - OTHER WRITE I/O**

The accumulated waiting time due to a write I/O performed for another thread during the execution of a package or DBRM.

**Background and Tuning Information**

If the value in this field is high, the problem could be I/O contention. The application is accessing data from a busy data set, volume, or control unit and is continually being suspended. Consult the DBA and MVS systems programmer to resolve possible data set placement problems.

**Field Name:** QPACAWTW

This is an *exception* field.

**AVERAGE TIME - SERV.TASK SWITCH**
**Package Times - Class 8 - Suspensions**

The accumulated waiting time due to a synchronous execution unit switch to DB2 services from this thread during the execution of the package or DBRM.

**Background and Tuning Information**

OMEGAMON XE for DB2 PE might adjust this value if the thread was suspended when its performance data was gathered.

This value includes the waits because of an OPEN/CLOSE data set, SYSLGRNG update, HSM RECALL data set, DATASPACE MANAGER services, DEFINE, EXTEND, and DELETE data set, and AUTONOMOUS PROCEDURE. Preformatting of data sets is a common cause of service task suspensions.

**Field Name:** QPACAWTE

This is an exception field.

**AVERAGE TIME - ARCH.LOG(QUIESCE)**

The accumulated waiting time caused by processing ARCHIVE LOG(QUIESCE) commands during the execution of the package or DBRM. This number represents the amount of time that an individual thread was suspended because of the command, not the time it took for the entire command to complete.

**Background and Tuning Information**

OMEGAMON XE for DB2 PE might adjust this value if the thread was suspended when its performance data was gathered.

Avoid issuing the -ARCHIVE LOG QUIESCE command during peak periods.

**Field Name:** QPACALOG

**AVERAGE TIME - DRAIN LOCK**

The accumulated waiting time due to a drain lock.

**Field Name:** QPACAWDR

**AVERAGE TIME - CLAIM RELEASE**

The accumulated waiting time for a drain waiting for claims to be released during the execution of the package or DBRM.

**Background and Tuning Information**

OMEGAMON XE for DB2 PE might adjust this value if the thread was suspended when its performance data was gathered.

**Field Name:** QPACAWCL

**AVERAGE TIME - PAGE LATCH**

The accumulated waiting time caused by a page latch contention.

**Field Name:** QPACAWTP

**AVERAGE TIME - NOTIFY MESSAGES**

The accumulated elapsed waiting time due to suspensions caused by sending notify messages to other members in the data sharing group. Messages are sent, for example, when database descriptors are changed due to DDL.
This value is only calculated if accounting class 8 is active and DB2 is a member of a DB2 data sharing group.

Field Name: QPACAWTG

AVERAGE TIME - GLOBAL CONTENTION

The accumulated waiting time caused by the suspension of IRLM lock requests due to global lock contentions in a data sharing environment that require intersystem communication to resolve.

Field Name: APGCSUST

AVERAGE TIME - TCP/IP LOB XML

The accumulated wait time for TCP/IP LOB and XML materialization while running this package or DBRM.

Field Name: QPACALBW

AVERAGE TIME - ACCELERATOR

The accumulated wait time for requests to an accelerator while executing this package.

Field Name: QPACAACW

AVERAGE TIME - PQ SYNCHRONIZATION

The accumulated time waiting for parallel query processing to synchronize between parent and child tasks.

Field Name: APPQSST

AVERAGE TIME - TOTAL CL8 SUSPENS.

The waiting time for the package or DBRM due to class 8 suspensions.

Field Name: ADTSUSTP

This is an exception field.

AVG.EV - LOCK/LATCH

This field depends on the DB2 version that is installed:

- **DB2 10 or later**: The sum of the number of wait trace events processed for waits for local contention for locks and the number of wait trace events processed for waits for latch contention while executing this package.
- **Prior to DB2 10**: The number of wait trace events processed for lock/latch while executing this package.

Note: The internally defined field adjusts the original DB2 value. DB2 counts each event twice, one for the entry and one for the exit.

Field Name: ADLLPSSC

This is an exception field.

AVG.EV - IRLM LOCK+LATCH

The number of wait trace events processed for waits for lock while executing this package.

Note: The internally defined field adjusts the original DB2 value. DB2 counts each event twice, one for the entry and one for the exit.
Package Times - Class 8 - Suspensions

Field Name: ADLBPS

**AVG.EV - DB2 LATCH**

The number of wait trace events processed for page latch contention while executing this package.

*Note:* The internally defined field adjusts the original DB2 value. DB2 counts each event twice, one for the entry and one for the exit.

Field Name: ADLAPSS

**AVG.EV - SYNCHRONOUS I/O**

The total number of synchronous I/O suspensions under this thread during the execution of the package or DBRM.

Field Name: ADIOPSS

**AVG.EV - OTHER READ I/O**

The total number of suspensions due to a read I/O performed under a thread other than the one being reported.

Field Name: ADARPSS

**AVG.EV - OTHER WRITE I/O**

The total number of suspensions due to a write I/O performed under a thread other than this one during the execution of a package or DBRM.

Field Name: ADAWPSS

**AVG.EV - SERV.TASK SWITCH**

The total number of suspensions due to a synchronous execution unit switch to DB2 services during the execution of the package or DBRM.

Field Name: ADSTPSS

**AVG.EV - ARCH.LOG(QUIESCE)**

The total number of suspensions caused by processing ARCHIVE LOG(QUIESCE) commands during the execution of the package or DBRM.

Field Name: ADALPSS

**AVG.EV - ARCHIVE LOG READ**

The number of wait trace events processed for archive reads, active reads, and active log prefetch reads while running this package or DBRM.

Field Name: ADLRPSS

**AVG.EV - DRAIN LOCK**

The total number of suspensions due to drain lock processing during the execution of the package or DBRM.

Field Name: ADDRPS

**AVG.EV - CLAIM RELEASE**

The total number of suspensions until the claims are released during the execution of the package or DBRM.

Field Name: ADCMPSS

**AVG.EV - PAGE LATCH**
Package Times - Class 8 - Suspensions

The total number of suspensions due to page latch contentions during the execution of the package or DBRM.

**Field Name:** ADPGPSSC

**AVG.EV - NOTIFY MESSAGES**

The number of suspensions due to messages being sent to other members in the data sharing group. This value is calculated only if accounting class 8 is active and DB2 is a member of a data sharing group.

**Field Name:** ADNOPSSC

**AVG.EV - GLOBAL CONTENTION**

The total number of suspensions during global lock contention. This value is calculated only if accounting class 8 is active and DB2 is a member of a data sharing group.

**Field Name:** ADGCPSSC

**AVG.EV - TCP/IP LOB XML**

The number of wait trace events processed for waits for TCP/IP LOB and XML materialization while running this package or DBRM.

**Field Name:** ADLMPSSC

**AVG.EV - ACCELERATOR**

The total number of suspensions due to a request to an accelerator during the execution of the package.

**Field Name:** ADAAPSSC

**AVG.EV - PQ SYNCHRONIZATION**

The number of times the parallel query processing suspended because it was waiting for the synchronization of the parent/child.

**Field Name:** APPQSSC

**AVG.EV - TOTAL CL8 SUSPENS.**

The number of all types of class 8 suspensions.

**Field Name:** ADTSUSCP

This is an *exception* field.

**TIME/EVENT - LOCK/LATCH**

This field depends on the DB2 version that is installed:

- **DB2 10 or later:** The sum of the number of wait trace events processed for waits for local contention for locks and the number of wait trace events processed for waits for latch contention while executing this package.

- **Prior to DB2 10:** The number of wait trace events processed for lock/latch while executing this package.

**Note:** The internally defined field adjusts the original DB2 value. DB2 counts each event twice, one for the entry and one for the exit.

**Field Name:** AALLPTMC

**TIME/EVENT - IRLM LOCK+LATCH**
Package Times - Class 8 - Suspensions

The number of wait trace events processed for waits for lock while executing this package.

Note: The internally defined field adjusts the original DB2 value. DB2 counts each event twice, one for the entry and one for the exit.

Field Name: AALOPTMC

TIME/EVENT - DB2 LATCH

The number of wait trace events processed for page latch contention while executing this package.

Note: The internally defined field adjusts the original DB2 value. DB2 counts each event twice, one for the entry and one for the exit.

Field Name: AALAPTMC

TIME/EVENT - SYNCHRONOUS I/O

The synchronous I/O suspension time per event.

Field Name: AAIOPTMC

TIME/EVENT - OTHER READ I/O

Any other read time per event.

Field Name: AAARPTMC

TIME/EVENT - OTHER WRITE I/O

Any other write time per event.

Field Name: AAWPTMC

TIME/EVENT - SERV.TASK SWITCH

The synchronous execution service time per event.

Field Name: AASTPTMC

TIME/EVENT - ARCH.LOGLG (QUIESCE)

The archive log time per event.

Field Name: AAALPTMC

TIME/EVENT - ARCHIVE LOG READ

The archive read suspension time per event.

Field Name: AALRPTMC

TIME/EVENT - DRAIN LOCK

The drain lock time per event.

Field Name: AADRPTMC

TIME/EVENT - CLAIM RELEASE

The claim release time per event.

Field Name: AACMPTMC

TIME/EVENT - PAGE LATCH

The page latch time per event.

Field Name: AAPGPTMC
Package Times - Class 8 - Suspensions

TIME/EVENT - NOTIFY MESSAGES
The notify messages time per event.
Field Name: AANOPTMC

TIME/EVENT - GLOBAL CONTENTION
The global contention time per event.
Field Name: AAGCPTMC

TIME/EVENT - TCP/IP LOB XML
The accumulated wait time for TCP/IP LOB and XML materialization while running this package or DBRM.
Field Name: QPACALBW

TIME/EVENT - ACCELERATOR
The accelerator waiting time per event during the execution of the package
Field Name: AAAAPTMC

TIME/EVENT - PQ SYNCHRONIZATION
The average wait time for a package for parallel query processing to synchronize between parent and child tasks.
Field Name: APPQSTMC

TIME/EVENT - TOTAL CL8 SUSPENS.
The class 8 time per event.
Field Name: AATOTSTP
Package Times - Class 7

This topic shows detailed information about “Accounting - Package Times - Class 7”.

This block shows the class 7 application times at package level.

For formatting reasons, OMEGAMON XE for DB2 PE shows different labels for report and trace. The following example shows both layouts, the report on the left, and the trace layout on the right.

**Accounting - Package Times - Class 7**

The field labels shown in the following sample layout of “Accounting - Package Times - Class 7” are described in the following section.

### Report:

<table>
<thead>
<tr>
<th>PACKAGE</th>
<th>TIMES</th>
<th>DSNTEP2</th>
<th>TIMES</th>
</tr>
</thead>
<tbody>
<tr>
<td>ELAP-CL7 TIME-AVG</td>
<td>0.019590</td>
<td></td>
<td>ELAPSED TIME - CL7</td>
</tr>
<tr>
<td>CP CPU TIME</td>
<td>0.002826</td>
<td>CP CPU TIME</td>
<td>0.015513</td>
</tr>
<tr>
<td>AGENT</td>
<td>0.002826</td>
<td>AGENT</td>
<td>0.015513</td>
</tr>
<tr>
<td>PAR.TASKS</td>
<td>0.000000</td>
<td>PAR.TASKS</td>
<td>0.000000</td>
</tr>
<tr>
<td>SE CPU TIME</td>
<td>0.000000</td>
<td>SE CPU TIME</td>
<td>0.000000</td>
</tr>
<tr>
<td>SUSPENSION-CL8</td>
<td>0.016115</td>
<td>SUSPENSION-CL8</td>
<td>0.000191</td>
</tr>
<tr>
<td>AGENT</td>
<td>0.016115</td>
<td>AGENT</td>
<td>0.000191</td>
</tr>
<tr>
<td>PAR.TASKS</td>
<td>0.000000</td>
<td>PAR.TASKS</td>
<td>0.000000</td>
</tr>
<tr>
<td>NOT ACCOUNTED</td>
<td>0.000649</td>
<td>NOT ACCOUNTED</td>
<td>3:16.199777</td>
</tr>
<tr>
<td>AVG.DB2 ENTRY/EXIT</td>
<td>N/P</td>
<td>CP CPU SU</td>
<td>208</td>
</tr>
<tr>
<td>DB2 ENTRY/EXIT</td>
<td>N/P</td>
<td>AGENT</td>
<td>208</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PAR.TASKS</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SE CPU SU</td>
<td>0</td>
</tr>
</tbody>
</table>

**ELAP-CL7 TIME-AVG**

The total elapsed time for executing the package or DBRM.

**Field Name:** QPACSCT

**CP CPU TIME**

The class 7 CPU time spent by the package or DBRM. It indicates:

- The TCB time
- The accumulated CPU time for processing parallel tasks. This is valid for query CP parallelism, sysplex query parallelism, and parallel tasks generated by utilities.

In sysplex query parallelism, only CPU times of parallel tasks, running on the same member of the sysplex group as the originating task, are included.

This time does not include the CPU time consumed on an IBM specialty engine.

**Field Name:** ADCPUTP

This is an *exception* field.

**AGENT**
Package Times - Class 7

The class 7 CPU time for all executions of the package or DBRM. This time does not include the:

- Class 7 time for parallel tasks
- CPU time that is consumed on an IBM specialty engine

Field Name: QPACTJST

This is an exception field.

AGENT - PAR.TASKS

The accumulated time for the package or DBRM to process parallel tasks. These tasks can be query CP, sysplex query, utility parallel tasks, or rollup autonomous tasks.

In sysplex query parallelism, the accumulated time reflects only parallel tasks running on the same DB2 subsystem as the originating task.

In case of rolled-up data, it is the sum of all CPU times, of originating and parallel tasks.

This time does not include CPU time consumed on an IBM specialty engine.

Field Name: ADCPCL7T

SE CPU TIME

The total CPU time for all executions of this package or DBRM that was consumed on an IBM specialty engine.

Note: All CPU times of an IBM Specialty Engine (SE) that are reported in DB2 trace records are already normalized by DB2 to the speed of the purpose processor.

Field Name: APACC7Z

SUSPENSION-CL8

The waiting time for the package or DBRM due to class 8 suspensions.

Field Name: ADTSUSTP

This is an exception field.

SUSPENSION-CL8 - AGENT

The class 8 suspension time for executing the package or DBRM. In query or utility parallelism, this does not include the class 8 time for parallel tasks.

Field Name: ADTCBCL8

SUSPENSION-CL8 - PAR.TASKS

The sum of the suspension times of the parallel tasks for the package or DBRM. The tasks can be query CP or sysplex query parallel tasks, tasks generated by utilities, or roll-up autonomous tasks.

In case of rolled-up data, it is the sum of all suspension times, of originating and parallel tasks.

Field Name: ADCPCL8T

NOT ACCOUNTED
Package Times - Class 7

The total unaccounted time in DB2 due to the execution of the package or DBRM. In query CP and sysplex query parallelism, it is the unaccounted time of the originating task only.

In case of rolled-up data, it is the unaccounted time of all tasks, of originating and parallel tasks.

Field Name: ADNACLA7T

This is an exception field.

AVG.DB2 ENTRY/EXIT

The number of DB2 entries or exits processed during the execution of the package or DBRM.

In Accounting reports this is shown twice; as a total and as an average.

Field Name: QPACARNA

CP CPU SU

The CPU service units for a package or DBRM. It indicates:

- The service units for the TCB time.
- The accumulated service units for processing parallel tasks if query CP or sysplex query parallelism is exploited.

These CPU service units do not include the service units that were consumed on an IBM specialty engine.

Field Name: ADSUCPU7

CP CPU SU - AGENT

The CPU service units for a package or DBRM. These CPU service units do not include the service units that were consumed on an IBM specialty engine.

Field Name: ADSUTCB7

CP CPU SU - PAR.TASKS

The CPU service units accumulated for a package or DBRM for processing parallel tasks. These tasks can be query CP or sysplex query parallel tasks, or roll-up autonomous tasks.

These service units do not include service units consumed on an IBM specialty engine.

Field Name: ADSUCPP7

SE CPU SU

The total CPU service units for all executions of this package or DBRM that were consumed on an IBM specialty engine.

Field Name: ADSUC7Z
Query Parallelism

This topic shows detailed information about “Accounting - Query Parallelism”.

If a query exploits query CP (central processor) parallelism or sysplex query parallelism, several tasks (called parallel tasks) perform the work. For each of these tasks an accounting record is generated, which contains counters and timers pertinent to the work performed by the particular task. In addition, an accounting record is created that contains the details on nonparallel work within the thread as well as data related to parallel work.

For formatting reasons, OMEGAMON XE for DB2 PE shows different labels for report and trace. The following example shows both layouts, the report on the left, and the trace layout on the right.

Accounting - Query Parallelism

The field labels shown in the following sample layout of “Accounting - Query Parallelism” are described in the following section.

<table>
<thead>
<tr>
<th>Report:</th>
<th>Trace:</th>
</tr>
</thead>
<tbody>
<tr>
<td>QUERY PARALLELISM</td>
<td>QUERY PARALLELISM</td>
</tr>
<tr>
<td>AVERAGE</td>
<td>TOTAL</td>
</tr>
<tr>
<td>MAXIMUM DEGREE-ESTIMATED</td>
<td>0.00</td>
</tr>
<tr>
<td>MAXIMUM DEGREE-PLANNED</td>
<td>0.00</td>
</tr>
<tr>
<td>MAXIMUM DEGREE-EXECUTED</td>
<td>0.00</td>
</tr>
<tr>
<td>MAXIMUM MEMBERS USED</td>
<td>N/A</td>
</tr>
<tr>
<td>GROUPS EXECUTED</td>
<td>0.00</td>
</tr>
<tr>
<td>RAN AS PLANNED</td>
<td>0.00</td>
</tr>
<tr>
<td>RAN REDUCED-STORAGE</td>
<td>0.00</td>
</tr>
<tr>
<td>RAN REDUCED-NEGOTIATION</td>
<td>0.00</td>
</tr>
<tr>
<td>SEQ-CURSOR</td>
<td>0.00</td>
</tr>
<tr>
<td>SEQ-NO ESA SORT</td>
<td>0.00</td>
</tr>
<tr>
<td>SEQ-NO BUFFER</td>
<td>0.00</td>
</tr>
<tr>
<td>SEQ-ENCLAVE SERVICES</td>
<td>N/A</td>
</tr>
<tr>
<td>SEQ-AUTONOMOUS PROCEDURE</td>
<td>0.00</td>
</tr>
<tr>
<td>SEQ-NEGOTIATION</td>
<td>0.00</td>
</tr>
<tr>
<td>ONE DB2-COORDINATOR + NO</td>
<td>0.00</td>
</tr>
<tr>
<td>ONE DB2-ISOLATION LEVEL</td>
<td>0.00</td>
</tr>
<tr>
<td>ONE DB2-OC1 TEMPORARY TABLE</td>
<td>0.00</td>
</tr>
<tr>
<td>MEMBER SKIPPED (%)</td>
<td>N/C</td>
</tr>
<tr>
<td>DISABLED BY RLF</td>
<td>0.00</td>
</tr>
<tr>
<td>REFORM PARAL-CONFIG</td>
<td>0.00</td>
</tr>
<tr>
<td>REFORM PARAL-NO BUF</td>
<td>0.00</td>
</tr>
</tbody>
</table>

MAXIMUM DEGREE-ESTIMATED (MAX DEGREE-ESTIMATE)

The maximum parallel-group estimated degree (DB2 field: QXMAXESTIDG). It is the bind time estimated degree based on the cost formula. If the parallel group contains a host variable or parameter marker, bind time will estimate the parallel-group degree based on a valid assumption value.

Field Name: AXMESTIDG

MAXIMUM DEGREE-PLANNED (MAX DEGREE-PLANNED)

The maximum parallel-group planned degree (DB2 field: QXMAXPLANDG). It is the ideal parallel-group degree obtained at execution time after the host variable or parameter marker value is "plug-in" and before the buffer pool negotiation and the system negotiation are performed.

Field Name: AXMPLANDG

MAXIMUM DEGREE-EXECUTED (MAX DEGREE-EXECUTED)

The maximum degree of parallelism executed among all parallel groups to indicate the extent to which queries were processed in parallel.
Query Parallelism

Field Name: QXMAXDEG

MAXIMUM MEMBERS USED (MAXIMUM MEMBERS)

The maximum number of DB2 members that participated in the processing of a query.

Field Name: AMAXMEMB

MAXIMUM DEGREE

The maximum degree of parallelism executed among all parallel groups to indicate the extent to which queries were processed in parallel.

Field Name: QXMAXDEG

GROUPS EXECUTED

The total number of parallel groups executed.

Field Name: QXTOTGRP

RAN AS PLANNED

The total number of parallel groups that executed in the planned parallel degree. This field is incremented by one for each parallel group that executed in the planned degree of parallelism (as determined by DB2).

Field Name: QXNORGRP

RAN REDUCED-STORAGE (REDUCED-STORAGE)

The total number of parallel groups that did not reach the planned parallel degree because of a lack of storage space or contention on the buffer pool. The exception field name is QXREDGRP.

Background and Tuning Information

If this field is not 0, increase the size of the current buffer pool using the ALTER BUFFERPOOL command or use the ALTER TABLESPACE command to assign table spaces accessed by this query to a different buffer pool.

Field Name: QXREDGRP

This is an exception field.

RAN REDUCED-NEGOTIATION (REDUCED-NEGOTIATE)

The number of parallel-group degrees that is reduced because of the system negotiation result of the system stress level (DB2 field: QXSTOREDGRP).

Field Name: AXREDPGD

SEQ - CURSOR

The total number of parallel groups that fell back to sequential mode due to a cursor that can be used by UPDATE or DELETE.

Field Name: QXDEGCUR

SEQ - NO ESA SORT (SEQ - NO ESA)

The total number of parallel groups that fell back to sequential mode due to a lack of ESA sort support.

Field Name: QXDEGESA
SEQ - NO BUFFER (SEQ - NO BUF)
The total number of parallel groups that fell back to sequential mode due to a storage shortage or contention on the buffer pool.
The exception field name is QXDEGBUF.
Field Name: QXDEGBUF

SEQ - ENCLAVE SERVICES (SEQ - ENCL.SER)
The total number of parallel groups that executed in sequential mode due to the unavailability of MVS/ESA enclave services.
Field Name: QXDEGENC
This is an exception field.

SEQ - AUTONOMOUS PROCEDURE (SEQ - AUTONOM.PROC)
The total number of parallel groups that fell back to sequential mode under an autonomous procedure.
Field Name: QXDEGAT

SEQ - NEGOTIATION
The number of parallel groups is degenerated to sequential because of the system negotiation result of system stress level (DB2 field: QXSTODGNGRP).
Field Name: AXDEGPGD

ONE DB2-COORDINATOR = NO (ONE DB2 COOR=N)
The total number of parallel groups executed on a single DB2 subsystem due to the COORDINATOR subsystem value being set to NO. When the statement was bound, the COORDINATOR subsystem value was set to YES. This situation can also occur when a package or plan is bound on a DB2 subsystem with COORDINATOR=YES, but is run on a DB2 subsystem with COORDINATOR=NO.
Field Name: QXCOORNO

ONE DB2-ISOLATION LEVEL (ONE DB2 ISOLAT)
The total number of parallel groups executed on a single DB2 subsystem due to repeatable-read or read-stability isolation.
Field Name: QXISISRR

ONE DB2-DCL TEMPORARY TABLE (ONE DB2 DCL TTABLE)
The number of parallel groups in a query block that were downgraded to CPU parallelism because they referenced a UDF and a declared temporary table was detected at execution time.
DB2 enforces execution on a single DB2 (CPU parallelism), in this instance, because it cannot determine at incremental bind time for the statement whether the UDF will reference the declared temporary table. Other parallel groups in the same statement are not necessarily downgraded.
Field Name: QXDEGDTT

MEMBER SKIPPED (%) (MEMB SKIPPED(%) )
The percentage of parallel groups that were not distributed over the data sharing group, as originally planned at bind time, because one or more
Query Parallelism

DB2 members did not have enough buffer pool storage. This only applies to parallel groups that were intended to run in sysplex query parallelism.

This percentage is to indicate a lack of buffers at a member. It is only increased when the buffer pool is defined to allow for parallelism. For example, if VXPSEQT=0 on an assistant, DB2 does not send parallel work there, and the percentage is not increased.

**Field Name:** AXXCRAT

**DISABLED BY RLF**

The number of threads where at least one dynamic SQL statement was disabled by the Resource Limit Facility (RLF).

**Field Name:** ADPARDNR

**REFORM PARAL-CONFIG**

The total number of parallel groups where DB2 reformulated the parallel portion of the access path because of a change in the number of active members, or because of a change of processor models on which they run, from bind time to run time. This counter is incremented only on the parallelism coordinator at run time.

**Field Name:** QXREPOP1

**REFORM PARAL-NO BUF**

The total number of parallel groups in which DB2 reformulated the parallel portion of the access path because there were insufficient buffer-pool resources. This counter is incremented only at the parallelism coordinator at run time.

**Field Name:** QXREPOP2
Resource Limit Facility

This topic shows detailed information about “Accounting - Resource Limit Facility”.

This block shows information about the Resource Limit Facility (RLF), which prevents dynamic manipulative SQL statements from exceeding specified time limits.

The following example shows both layouts, the report layout followed by the trace layout.

Accounting - Resource Limit Facility

The field labels shown in the following sample layout of “Accounting - Resource Limit Facility” are described in the following section.

Report:

<table>
<thead>
<tr>
<th>RESOURCE LIMIT TYPE</th>
<th>#OCCUR</th>
<th>AVERAGE CPU SECONDS</th>
<th>HIGHEST CPU SECONDS</th>
</tr>
</thead>
<tbody>
<tr>
<td>INFINITE LIMIT</td>
<td>1</td>
<td>5.000000</td>
<td>5.000000</td>
</tr>
</tbody>
</table>

Trace:

```
---- RESOURCE LIMIT FACILITY

TYPE: N/P  TABLE ID: N/P  SERV.UNITS: N/P  CPU SECONDS: 0.000000  MAX CPU SEC: N/P

RESOURCE LIMIT TYPE (TYPE)

The resource limit type. This is taken from QTXAPREC and can be:
• AUTHID/PLAN
• AUTHID ANY PLAN
• PLAN ANY AUTHID
• BLANK AUTHID & PLAN
• INSTALL NO ENTRY
• INSTALL I/O ERROR
• NO LIMIT - SYSADM/SYSOPR
• AUTHID/COLLECTION/PACKAGE
• AUTHID ANY PACKAGE
• AUTHID ANY COLLECTION
• AUTHID ANY PACKAGE/COLLECTION
• PACKAGE/COLLECTION ANY AUTHID
• ANY AUTHID/PACKAGE
• ANY AUTHID/COLLECTION
• ANY AUTHID/PACKAGE/COLLECTION

The following can be reported:

INFINITE LIMIT
Reported when QTXAPREC has any value except NO LIMIT - SYSADM/SYSOPR and QTXAILMT is on.

NO RUN OR ZERO LIMIT
Reported when QTXANRUN is on.

Field Name: ADRLFTYP

#OCCUR
```
Resource Limit Facility

The number of RLF occurrences.

Field Name: ASRLFOCC

AVERAGE CPU SECONDS

The number of CPU seconds used.

Field Name: ADRLFCPU

HIGHEST CPU SECONDS

The highest CPU seconds used in a successful DB2 internal call rather than in a single SQL call. Because there are usually many DB2 calls for each SQL call, this value could be quite small compared to the total CPU time used in the SQL call.

Only times for successful DB2 calls are used to determine the value of this field.

Field Name: ADRLFMAX

TABLE ID

The identifier of the resource limit specification table.

Field Name: QTXARLID

SERV.UNIT

The maximum number of CPU service units to be used. Normally, the value is not 0 if the RES LIMIT TYPE is LIMIT. A value of 0 indicates no limit.

Field Name: QTXASLMT
RID List

This topic shows detailed information about “Accounting - RID List”.

This block shows information about the Record identifier (RID) list.

The following example shows both layouts, the report on the left, and the trace layout on the right.

**Accounting - RID List**

The field labels shown in the following sample layout of “Accounting - RID List” are described in the following section.

<table>
<thead>
<tr>
<th>Report:</th>
<th>Trace:</th>
</tr>
</thead>
<tbody>
<tr>
<td>RID LIST</td>
<td>AVERAGE</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>USED</td>
<td>0.00</td>
</tr>
<tr>
<td>FAIL-NO STORAGE</td>
<td>0.00</td>
</tr>
<tr>
<td>FAIL-LIMIT EXCEEDED</td>
<td>0.00</td>
</tr>
<tr>
<td>INTERRUPTED-NO STORAGE</td>
<td>0.00</td>
</tr>
<tr>
<td>INTERRUPTED-LIMIT EXC.</td>
<td>0.00</td>
</tr>
<tr>
<td>OVERFLOWED-NO STORAGE</td>
<td>0.00</td>
</tr>
<tr>
<td>OVERFLOWED-LIMIT EXC.</td>
<td>0.00</td>
</tr>
<tr>
<td>SKIPPED-INDEX KNOWN</td>
<td>0.00</td>
</tr>
</tbody>
</table>

**USED**

The number of times RID list (also called RID pool) processing is used.

During RID (RECORD ID) list processing, DB2 uses an index to produce a list of candidate RIDs, which is called a RID list. The RID list can be sorted and intersected (ANDed) or unioned (ORed) with other RID lists before actually accessing the data pages. RID list processing is used for a single index (index access with list prefetch) or for multiple indexes (multiple index access), which is when the RID lists are ANDed and ORed.

This field is incremented once for a given table access when RID list processing is used for index access with list prefetch, for multiple index access, or for both. For multiple index access, if a final RID list is obtained through ANDing and ORing of RID lists, the counter is incremented once, even if not all indexes were used by the RIDs in the multiple index access.

**Background and Tuning Information**

A nonzero value in this field indicates that DB2 has used list prefetch. If this is the case, check the access path selection.

**Field Name:** QXMIAP

This is an exception field.

**FAIL-NO STORAGE**

The number of times DB2 detected that no storage was available to hold a list of RIDs during a given RID list process involving one index (single index access with list prefetch) or multiple indexes (multiple index access).

This field can be incremented during retrieval, sorting, ANDing, and ORing of RID lists for index access with list prefetch (single index). For
RID List

single index access, this field can only be incremented once per access. For multiple index access, it can be incremented for every index involved in the ANDing and ORing of RID lists.

Field Name: QXNSMIAP
This is an exception field.

FAIL-LIMIT EXCEEDED (FAIL-LIMIT EXC.)
The number of times DB2 detected that a RID list exceeded one or more internal limits during a given RID list (or RID pool) process involving one index (single index access with list prefetch) or multiple indexes (multiple index access). The internal limits include the physical limitation of the number of RIDs a RID list can hold and threshold values for the retrieval, ORing, and ANDing of RIDs.
For index access with list prefetch (single index), this field can only be incremented during RID list retrieval. For multiple index access, this field can be incremented during RID list retrieval, ANDing, and ORing. This counter reflects the number of times internal limits or threshold values were exceeded for the RID lists obtained directly from an index as well as for RID lists derived during the ANDing and ORing process.

Background and Tuning Information
Before you increase the RID list storage size, investigate the cause of the failure using the statistics record or the performance trace. You can specify the desired size for the RID list (within the range of 16 KB to 1000 MB) on the DB2 installation panel DSNTIFC.

Field Name: QXMRRMIAP
This is an exception field.

INTERRUPTED-NO STORAGE
The number of times a RID list append for a hybrid join was interrupted because no RID pool storage was available to hold the list of RIDs.

Field Name: QXHJINCS

INTERRUPTED-LIMIT EXC.
The number of times a RID list append for a hybrid join was interrupted because the number of RIDs exceeded one or more internal limits.

Field Name: QXHJINCT

OVERFLOWED-NO STORAGE
The number of times a RID list was overflowed to a work file because no RID pool storage was available to hold the list of RIDs.

Field Name: QXWFRIDS

OVERFLOWED-LIMIT EXC.
The number of times a RID list was overflowed to a work file because the number of RIDs exceeded one or more internal limits.

Field Name: QXWFRIDT

SKIPPED-INDEX KNOWN
The number of times a RID list retrieval for multiple index access was skipped because it was not necessary due to DB2 being able to predetermine the outcome of index ANDing or ORing.

Field Name: QXRSMIAP
ROWID

ROWID

This topic shows detailed information about “Accounting - ROWID”.

This block shows information about the row identifier (ROWID).

The following example shows both layouts, the report on the left, and the trace layout on the right.

**Accounting - ROWID**

The field labels shown in the following sample layout of “Accounting - ROWID” are described in the following section.

<table>
<thead>
<tr>
<th>Report</th>
<th>Trace</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROWID</td>
<td>ROWID</td>
</tr>
<tr>
<td>AVERAGE</td>
<td>TOTAL</td>
</tr>
<tr>
<td>TOTAL</td>
<td>TOTAL</td>
</tr>
<tr>
<td>DIRECT ACCESS</td>
<td>0.00</td>
</tr>
<tr>
<td>INDEX USED</td>
<td>0.00</td>
</tr>
<tr>
<td>TS SCAN USED</td>
<td>0.00</td>
</tr>
</tbody>
</table>

**DIRECT ACCESS (DIR ACCESS)**

The number of times that direct row access was successful.

**Field Name:** QXROIMAT

**INDEX USED**

The number of times that direct row access failed and an index was used to find a record.

**Background and Tuning Information**

This can happen, for example, when a REORG is performed between the read of the ROWID column and the use of the host variable in the WHERE clause of the SQL statement. This causes the RID value in the host variable to be incorrect.

**Field Name:** QXROIIDX

**TS SCAN USED (TS SCAN)**

The number of times that an attempt to use direct row access reverted to using a table-space scan because DB2 was unable to use a matching index scan.

**Background and Tuning Information**

Ideally, this value should be 0.

Table-space scans can happen, for example, when a REORG is performed between the read of the ROWID column and the use of the host variable in the WHERE clause of the SQL statement. This causes the RID value in the host variable to be incorrect. DB2 first tries a matching-index scan before using a table-space scan.

To avoid table space scans, you can force the access path of an unsuccessful direct row access to use a matching index scan on the primary-index key by adding PKCOL to the WHERE clause in the SQL statement. .... WHERE ROWIDCOL=:HVROWID AND PKCOL=:HVPK

**Field Name:** QXROITS
Service Units

This topic shows detailed information about “Accounting - Service Units”.

This block shows class 1 and class 2 CPU times as service units.

The following example shows both layouts, the report on the left, and the trace layout on the right.

Accounting - Service Units

The field labels shown in the following sample layout of “Accounting - Service Units” are described in the following section.

<table>
<thead>
<tr>
<th>Report</th>
<th>Trace</th>
</tr>
</thead>
<tbody>
<tr>
<td>AVERAGE SU</td>
<td>CLASS 1</td>
</tr>
<tr>
<td>CP CPU</td>
<td>15.50</td>
</tr>
<tr>
<td>AGENT</td>
<td>15.50</td>
</tr>
<tr>
<td>NONNESTED</td>
<td>15.50</td>
</tr>
<tr>
<td>STORED PRC</td>
<td>0.00</td>
</tr>
<tr>
<td>UDF</td>
<td>0.00</td>
</tr>
<tr>
<td>TRIGGER</td>
<td>0.00</td>
</tr>
<tr>
<td>PAR.TASKS</td>
<td>0.00</td>
</tr>
<tr>
<td>SECP CPU</td>
<td>0.00</td>
</tr>
<tr>
<td>SE CPU</td>
<td>15.50</td>
</tr>
<tr>
<td>NONNESTED</td>
<td>15.50</td>
</tr>
<tr>
<td>STORED PROC</td>
<td>0.00</td>
</tr>
<tr>
<td>UDF</td>
<td>0.00</td>
</tr>
<tr>
<td>TRIGGER</td>
<td>0.00</td>
</tr>
<tr>
<td>PAR.TASKS</td>
<td>0.00</td>
</tr>
</tbody>
</table>

CLASS 1: CP CPU

The class 1 CPU service units (in an application). It indicates:

- The TCB service units.
- The accumulated TCB service units for processing stored procedures if stored procedures are present.
- The accumulated CPU service units for processing parallel tasks. This is valid for query CP parallelism, sysplex query parallelism, and parallel tasks generated by utilities.

These CPU service units do not include the service units that were consumed on an IBM specialty engine.

Field Name: ADSUCPU1

CLASS 1: CP CPU - AGENT

The class 1 TCB service units (in an application). This field is derived from the TCB time and the conversion factor of the originating task.

Field Name: ADSUAGT1

CLASS 1: CP CPU - AGENT - NONNESTED

The TCB service units accumulated in nonnested activity.

Field Name: ADSUNN1

CLASS 1: CP CPU - AGENT - STORED PRC

The TCB service units accumulated in an application for stored procedures. This field is derived from the TCB time and the conversion factor of the originating task.

Field Name: ADSUTCS1
CLASS 1: CP CPU - AGENT - UDF

The TCB service units accumulated in an application for UDF.

Field Name: ADSUTCU1

CLASS 1: CP CPU - AGENT - TRIGGER

The number of TCB service units accumulated in DB2 used while executing under control of a trigger.

Field Name: ADSUTCT2

CLASS 1: CP CPU - PAR.TASKS

The sum of the CPU service units of the parallel tasks running in an application. These tasks can be query CP or sysplex query parallel tasks, parallel tasks produced by utilities, or roll-up autonomous tasks.

Field Name: ADSUCPP1

CLASS 1: CP CPU - SECP CPU

The accumulated CPU service units that ran on a standard CP for work eligible on an IBM specialty engine.

Field Name: ADSUZEL

CLASS 1: SE CPU

The sum of several accumulated CPU service units consumed while running on an IBM specialty engine in all environments. This field is derived from the TCB time and the conversion factor of the originating task. These service units are consumed when:

• Running stored procedure requests and triggers on the main application execution unit.
• Satisfying stored procedure requests processed in a DB2 stored procedure or WLM address space. SQLP times are included in this time if the SQLP was called on a nested task and was not invoked by the main application execution unit.
• Satisfying UDF requests processed in a DB2 stored procedure or WLM address space.
• Running triggers on a nested task.
• Running parallel tasks in an application which contains the accumulated CPU time used to satisfy UDF requests.

Note: All CPU service units of an IBM Specialty Engine (SE) that are reported in DB2 trace records are already normalized by DB2 to the speed of the purpose processor.

Field Name: ADSUC1Z

CLASS 1: SE CPU - NONNESTED

The class 1 CPU service units for nonnested activity on the main application task consumed while running on an IBM specialty engine. These service units ignore the CPU time consumed when running stored procedure requests, or triggers on the main application execution unit on an IBM specialty engine.

Field Name: ADSUSEN1

CLASS 1: SE CPU - STORED PROC

Service Units
Service Units

The accumulated and consumed service units for stored procedures on an
IBM specialty engine that consist of following parts:

- Service units processed in a DB2 stored procedure or WLM address
  space. SQLP service units are included if the SQLP was called on a
  nested task and was not invoked by the main application execution unit.
- Service units when running on the main application execution unit. As
  these stored procedures run entirely within DB2, this part of the service
  units counts for class 1 and class 2 time.

Field Name: ADSUSES1

CLASS 1: SE CPU - UDF

The accumulated CPU service units used to satisfy UDF requests processed
in a DB2 stored procedure or WLM address space while running on an
IBM specialty engine.

Field Name: ADSUSEU1

CLASS 1: SE CPU - TRIGGER

The accumulated CPU service units consumed on an IBM specialty engine
while running triggers on a nested task or on the main application
execution unit.

Field Name: ADSUSETR

CLASS 1: SE CPU - PAR.TASKS

The sum of the CPU service units of the parallel tasks running in an
application on an IBM specialty engine. These service units contain the
nonnested and consumed service units for stored procedures, UDFs, and
triggers.

Field Name: ADSUSEP1

CLASS 2: CP CPU

The class 2 service units (in DB2). It indicates:

- The TCB service units.
- The accumulated TCB service units for processing stored procedures if
  stored procedures are present.
- The accumulated CPU service units for processing parallel tasks. This is
  valid for query CP parallelism, sysplex query parallelism, and parallel
tasks produced by utilities.

These CPU service units do not include the service units that were
consumed on an IBM specialty engine.

Field Name: ADSUCPU2

CLASS 2: CP CPU - AGENT

The class 2 TCB service units (in DB2). This field is derived from the TCB
time and the conversion factor of the originating task.

Field Name: ADSUAGT2

CLASS 2: CP CPU - AGENT - NONNESTED

The number of class 2 service units accumulated in nonnested activity.

Field Name: ADSUNN2

CLASS 2: CP CPU - AGENT - STORED PRC
**Service Units**

The TCB service units accumulated in DB2 for stored procedures. This field is derived from the TCB time and the conversion factor of the originating task.

**Field Name:** ADSUTCS2

**CLASS 2: CP CPU - AGENT - UDF**

The TCB service units accumulated in DB2 for UDF.

**Field Name:** ADSUTCU2

**CLASS 2: CP CPU - AGENT - TRIGGER**

The number of TCB service units accumulated in DB2 used while executing under control of a trigger.

**Field Name:** ADSUTCT2

**CLASS 2: CP CPU - PAR.TASKS**

The sum of the CPU service units of the parallel tasks running in DB2. These tasks can be query CP or sysplex query parallel tasks, parallel tasks produced by utilities, or roll-up autonomous tasks.

These service units do not include service units consumed on an IBM specialty engine.

**Field Name:** ADSUCPP2

**CLASS 2: SE CPU**

The sum of the accumulated CPU service units consumed while running in DB2 on an IBM specialty engine due to CPU time spent:

- Nonnested on main application execution unit.
- On triggers on main application execution unit and nested tasks.
- Processing SQL statements issued by UDFs processed in a DB2 stored procedure or WLM address space.
- On stored procedures on main application execution unit and nested tasks processed in a DB2 stored procedure or WLM address space. SQLP times are included if the SQLP was called on a nested task and was not invoked by the main application execution unit.

**Field Name:** ADSUC2Z

**CLASS 2: SE CPU - NONNESTED**

The class 2 CPU service units for nonnested activity on the main application task consumed while running on an IBM specialty engine.

**Field Name:** ADSUSEN2

**CLASS 2: SE CPU - STORED PROC**

The accumulated and consumed service units for stored procedures on an IBM specialty engine that consist of following parts:

- Service units consumed in DB2, in a DB2 stored procedure, or WLM address space. SQLP service units are included if the SQLP was called on a nested task and was not invoked by the main application execution unit.
- Service units when running on the main application execution unit. As these stored procedures run entirely within DB2, this part of service units counts for class 1 and class 2 time.
Service Units

**Field Name:** ADSUSES2

**CLASS 2: SE CPU - UDF**

The accumulated and consumed service units for stored procedures on an IBM specialty engine that consist of following parts:

- Service units consumed in DB2, in a DB2 stored procedure, or WLM address space. SQLP service units are included if the SQLP was called on a nested task and was not invoked by the main application execution unit.
- Service units when running on the main application execution unit. As these stored procedures run entirely within DB2, this part of service units counts for class 1 and class 2 time.

This time is a subset of QWACSP_CLS1SE.

**Field Name:** ADSUSEU2

**CLASS 2: SE CPU - TRIGGER**

The accumulated CPU service units consumed on an IBM specialty engine while running triggers on a nested task or on the main application execution unit.

**Field Name:** ADSUSETR

**CLASS 2: SE CPU - PAR.TASKS**

The sum of the CPU service units of the parallel tasks running in DB2. These service units contain the nonnested and consumed service units for stored procedures, UDFs, and triggers.

**Field Name:** ADSUSEP2
**Stored Procedures**

**Stored Procedures**

This topic shows detailed information about “Accounting - Stored Procedures”.

This block shows information about stored procedure.

For formatting reasons, OMEGAMON XE for DB2 PE shows different labels for report and trace. The following example shows both layouts, the report on the left, and the trace layout on the right.

**Accounting - Stored Procedures**

The field labels shown in the following sample layout of “Accounting - Stored Procedures” are described in the following section.

<table>
<thead>
<tr>
<th>Report:</th>
<th>Trace:</th>
</tr>
</thead>
<tbody>
<tr>
<td>STORED PROCEDURES</td>
<td>STORED PROC.</td>
</tr>
<tr>
<td>AVERAGE</td>
<td>TOTAL</td>
</tr>
<tr>
<td>CALL STATEMENTS</td>
<td>0.00</td>
</tr>
<tr>
<td>ABENDED</td>
<td>0.00</td>
</tr>
<tr>
<td>TIMED OUT</td>
<td>0.00</td>
</tr>
<tr>
<td>REJECTED</td>
<td>0.00</td>
</tr>
</tbody>
</table>

**CALL STATEMENTS (CALL STMTS)**

The number of SQL CALL statements executed.

*Field Name:* QXCALL

**ABENDED**

The number of times a stored procedure terminated abnormally.

*Field Name:* QXCALLAB

**TIMED OUT**

The number of times an SQL call timed out waiting to be scheduled.

*Field Name:* QXCALLTO

**REJECTED**

The number of times an SQL CALL statement was rejected due to the procedure being in the STOP ACTION(REJECT) state.

*Field Name:* QXCALLRRJ
This topic shows detailed information about “Accounting - SQL DCL”.

This block shows information about SQL DCL (Data Control Language) declarations.

The following example applies to both, the report layout and the trace layout.

**Accounting - SQL DCL**

The field labels shown in the following sample layout of “Accounting - SQL DCL” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOCK TABLE</td>
<td>The number of LOCK TABLE statements executed.</td>
</tr>
<tr>
<td>GRANT</td>
<td>The number of GRANT statements executed.</td>
</tr>
<tr>
<td>REVOKE</td>
<td>The number of REVOKE statements executed.</td>
</tr>
<tr>
<td>SET CURR.SQLID</td>
<td>The number of SET CURRENT SQLID statements executed.</td>
</tr>
<tr>
<td>SET HOST VAR.</td>
<td>The number of SET HOST VARIABLE statements executed. The special register</td>
</tr>
<tr>
<td></td>
<td>that was retrieved is not tracked.</td>
</tr>
<tr>
<td>SET CURR.DEGREE</td>
<td>The number of SET CURRENT DEGREE statements executed.</td>
</tr>
<tr>
<td>SET RULES</td>
<td>The number of SET RULES statements executed.</td>
</tr>
<tr>
<td>SET CURR.PATH</td>
<td>The number of SET CURRENT PATH statements executed.</td>
</tr>
<tr>
<td>SET CURR.PREC</td>
<td>The number of SET CURRENT PRECISION statements executed.</td>
</tr>
<tr>
<td>CONNECT TYPE 1</td>
<td>The number of CONNECT TYPE 1 statements executed.</td>
</tr>
<tr>
<td>CONNECT TYPE 2</td>
<td>The number of CONNECT TYPE 2 statements executed.</td>
</tr>
<tr>
<td>SET CONNECTION</td>
<td>The number of SET CONNECTION statements executed.</td>
</tr>
<tr>
<td>RELEASE</td>
<td>The number of RELEASE statements executed.</td>
</tr>
<tr>
<td>CALL</td>
<td>The number of CALL statements executed.</td>
</tr>
<tr>
<td>ASSOC LOCATORS</td>
<td>The number of ASSOC LOCATORS statements executed.</td>
</tr>
<tr>
<td>ALLOC CURSOR</td>
<td>The number of ALLOC CURSOR statements executed.</td>
</tr>
<tr>
<td>HOLD LOCATOR</td>
<td>The number of HOLD LOCATOR statements executed.</td>
</tr>
<tr>
<td>FREE LOCATOR</td>
<td>The number of FREE LOCATOR statements executed.</td>
</tr>
<tr>
<td>DCL-ALL</td>
<td>The number of DCL-ALL statements executed.</td>
</tr>
</tbody>
</table>

**LOCK TABLE**

The number of LOCK TABLE statements executed.

**Field Name:** QXLOCK

**GRANT**

The number of GRANT statements executed.

**Field Name:** QXGRANT

**REVOKE**

The number of REVOKE statements executed.

**Field Name:** QXREVOK

**SET CURR.SQLID**

The number of SET CURRENT SQLID statements executed.

**Field Name:** QXSETSQL

**SET HOST VAR.**

The number of SET HOST VARIABLE statements executed. The special register that was retrieved is not tracked.

**Field Name:** QXSETHV

**SET CURR.DEGREE**
SQL DCL

The number of SET CURRENT DEGREE statements executed.
Field Name: QXSETCDG

SET RULES
The number of SET CURRENT RULES statements executed.
Field Name: QXSETCRL

SET CURR.PATH
The number of SET CURRENT PATH statements executed.
Field Name: QXSETPTH

SET CURR.PREC
The number of SET CURRENT PRECISION statements executed.
Field Name: QXSETCPR

CONNECT TYPE 1
The number of CONNECT type 1 statements executed.
Field Name: QXCON1

CONNECT TYPE 2
The number of CONNECT type 2 statements executed.
Field Name: QXCON2

SET CONNECTION
The number of SET CONNECTION statements executed.
Field Name: QXSETCON

RELEASE
The number of RELEASE statements executed.
Field Name: QXREL

CALL
The number of SQL CALL statements executed.
Field Name: QXCALL

ASSOC LOCATORS
The number of SQL ASSOCIATE LOCATORS statements executed.
Field Name: QXALOCL

ALLOC CURSOR
The number of SQL ALLOCATE CURSOR statements executed.
Field Name: QXALOCC

HOLD LOCATOR
The number of HOLD LOCATOR statements executed.
Field Name: QXHLDLOC

FREE LOCATOR
The number of times a FREE LOCATOR statement was issued.
Field Name: QXFRELOC

DCL-ALL

The total number of DCL statements executed.

Field Name: ASCDCL
SQL DDL

This topic shows detailed information about “Accounting - SQL DDL”.

This block shows information about SQL DDL (Data Definition Language) statements.

The following example applies to both, the report layout and the trace layout.

**Accounting - SQL DDL**

The field labels shown in the following sample layout of “Accounting - SQL DDL” are described in the following section.

<table>
<thead>
<tr>
<th>SQL DDL</th>
<th>CREATE</th>
<th>DROP</th>
<th>ALTER</th>
</tr>
</thead>
<tbody>
<tr>
<td>TABLE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>CRT TTABLE</td>
<td>0</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>DCL TTABLE</td>
<td>0</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>AUX TABLE</td>
<td>0</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>INDEX</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TABLESPACE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DATABASE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>STOGROUP</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SYNONYM</td>
<td>0</td>
<td>0</td>
<td>N/A</td>
</tr>
<tr>
<td>VIEW</td>
<td>0</td>
<td>0</td>
<td>513</td>
</tr>
<tr>
<td>ALIAS</td>
<td>0</td>
<td>0</td>
<td>N/A</td>
</tr>
<tr>
<td>PACKAGE</td>
<td>N/A</td>
<td>0</td>
<td>N/A</td>
</tr>
<tr>
<td>PROCEDURE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TRIGGER</td>
<td>0</td>
<td>0</td>
<td>N/A</td>
</tr>
<tr>
<td>DIST TYPE</td>
<td>0</td>
<td>0</td>
<td>N/A</td>
</tr>
<tr>
<td>SEQUENCE</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TRUST. CTX</td>
<td>521</td>
<td>522</td>
<td>523</td>
</tr>
<tr>
<td>ROLE</td>
<td>531</td>
<td>532</td>
<td>N/A</td>
</tr>
<tr>
<td>JAR</td>
<td>N/A</td>
<td>N/A</td>
<td>543</td>
</tr>
<tr>
<td>MASK/PERM</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>VARIABLE</td>
<td>0</td>
<td>0</td>
<td>N/A</td>
</tr>
<tr>
<td>TOTAL</td>
<td>1052</td>
<td>1054</td>
<td>1579</td>
</tr>
<tr>
<td>TRUNC TBL</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RENAME TBL</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RENAME IX</td>
<td>551</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMENT ON</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LABEL ON</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**CREATE TABLE**

The number of CREATE TABLE statements executed.

*Field Name:* QXCRTAB

**CREATE CRT TTABLE**

The number of CREATE GLOBAL TEMPORARY TABLE statements executed.

*Field Name:* QXCRGTT

**CREATE DCL TTABLE**

The number of DECLARE GLOBAL TEMPORARY TABLE statements executed.

*Field Name:* QXDCLGTT

**CREATE AUX TTABLE**

5-182  OMEGAMON XE for DB2 PE & PM: Report Reference
SQL DDL

The number of CREATE AUXILIARY TABLE statements executed.
Field Name: QXCRATB

CREATE INDEX
The number of CREATE INDEX statements executed.
Field Name: QXCRINX

CREATE TABLESPACE
The number of CREATE TABLESPACE statements executed.
Field Name: QXCTABS

CREATE DATABASE
The number of CREATE DATABASE statements executed.
Field Name: QXCRDAB

CREATE STOGROUP
The number of CREATE STOGROUP statements executed.
Field Name: QXCRSTG

CREATE SYNONYM
The number of CREATE SYNONYM statements executed.
Field Name: QXCRSYN

CREATE VIEW
The number of CREATE VIEW statements executed.
Field Name: QXDEFVU

CREATE ALIAS
The number of CREATE ALIAS statements executed.
Field Name: QXCRALS

CREATE PROCEDURE
The number of CREATE PROCEDURE statements issued.
Field Name: QXCRPRO

CREATE FUNCTION
The number of CREATE FUNCTION statements executed.
Field Name: QXCRUDF

CREATE TRIGGER
The number of CREATE TRIGGER statements executed.
Field Name: QXCTRIG

CREATE DIST TYPE
The number of CREATE DISTINCT TYPE statements executed.
Field Name: QXCDIST

CREATE SEQUENCE
The number of CREATE SEQUENCE statements.
Field Name: QXCRESEQ
CREATE TRUST. CTX
The number of CREATE TRUSTED CONTEXT statements issued.
Field Name: QXCRCTX
CREATE ROLE
The number of CREATE ROLE statements executed.
Field Name: QXCRROL
CREATE MASK/PERM
The number of CREATE MASK and CREATE PERMISSION statements executed.
Field Name: QXCREMP
CREATE VARIABLE
The number of CREATE VARIABLE statements.
Field Name: QXCRTSV
TOTAL CREATE
The number of SQL CREATE statements executed.
Field Name: ASTOTCRT
This is an exception field.
DROP TABLE
The number of DROP TABLE statements executed.
Field Name: QXDRPTA
This is an exception field.
DROP INDEX
The number of DROP INDEX statements executed.
Field Name: QXDRPIX
This is an exception field.
DROP TABLESPACE
The number of DROP TABLESPACE statements executed.
Field Name: QXDRPTS
This is an exception field.
DROP DATABASE
The number of DROP DATABASE statements executed.
Field Name: QXDRPDB
This is an exception field.
DROP STOGROUP
The number of DROP STOGROUP statements executed.
Field Name: QXDRPST
This is an exception field.
## SQL DDL

### DROP SYNONYM

The number of DROP SYNONYM statements executed.

**Field Name:** QXDRPSY

This is an *exception* field.

### DROP VIEW

The number of DROP VIEW statements executed.

**Field Name:** QXDRPVU

This is an *exception* field.

### DROP ALIAS

The number of SQL DROP ALIAS statements executed.

**Field Name:** QXDRPAL

This is an *exception* field.

### DROP PACKAGE

The number of SQL DROP PACKAGE statements executed.

**Field Name:** QXDRPPKG

This is an *exception* field.

### DROP PROCEDURE

The number of DROP PROCEDURE statements executed.

**Field Name:** QXDRPRR

### DROP FUNCTION

The number of DROP FUNCTION statements executed.

**Field Name:** QXDRPFN

### DROP TRIGGER

The number of DROP TRIGGER statements executed.

**Field Name:** QXDRPTR

### DROP DIST TYPE

The number of DROP DISTINCT TYPE statements executed.

**Field Name:** QXDDIST

### DROP SEQUENCE

The number of DROP SEQUENCE statements.

**Field Name:** QXDROSEQ

### DROP TRUST. CTX

The number of DROP TRUSTED CONTEXT statements issued.

**Field Name:** QXDRPCTX

### DROP ROLE

The number of DROP ROLE statements issued.

**Field Name:** QXDRPROL
SQL DDL

DROP MASK/PERM

The number of DROP MASK and DROP PERMISSION statements executed.

Field Name: QXDRPMP

DROP VARIABLE

The number of DROP VARIABLE statements.

Field Name: QXDRPSV

TOTAL DROP

The number of SQL DROP statements executed.

Field Name: ASTOTDRP

This is an exception field.

ALTER TABLE

The number of ALTER TABLE statements executed.

Field Name: QXALTTA

This is an exception field.

ALTER INDEX

The number of ALTER INDEX statements executed.

Field Name: QXALTIX

This is an exception field.

ALTER TABLESPACE

The number of ALTER TABLESPACE statements executed.

Field Name: QXALTTS

This is an exception field.

ALTER DATABASE

The number of ALTER DATABASE statements executed.

Field Name: QXALDAB

This is an exception field.

ALTER STOGROUP

The number of ALTER STOGROUP statements executed.

Field Name: QXALTST

This is an exception field.

ALTER VIEW

The number of ALTER VIEW statements issued.

Field Name: QXALTVW

ALTER PROCEDURE

The number of ALTER PROCEDURE statements executed.

Field Name: QXALPRO

ALTER FUNCTION
The number of ALTER FUNCTION statements executed.
Field Name: QXALUDF

ALTER SEQUENCE
The number of ALTER SEQUENCE statements.
Field Name: QXALTSEQ

ALTER TRUST_CTX
The number of ALTER TRUSTED CONTEXT statements issued.
Field Name: QXALTCTX

ALTER ROLE
The number of ALTER JAR statements issued.
Field Name: QXALTJIR

ALTER MASK/PERM
The number of ALTER MASK and ALTER PERMISSION statements executed.
Field Name: QXALTMP

TOTAL ALTER
The number of SQL ALTER statements executed.
Field Name: ASTOTALT
This is an exception field.

TRUNC TBL
The number of TRUNCATE TABLE statements issued.
Field Name: QXTRTBL

RENAME TBL
The number of RENAME TABLE statements executed.
Field Name: QXRNTAB

RENAME IX
The number of RENAME INDEX statements issued.
Field Name: QXRNIX

COMMENT ON
The number of COMMENT ON statements executed.
Field Name: QXCMTON

LABEL ON
The number of LABEL ON statements executed.
Field Name: QXLABON
**SQL DML**

This topic shows detailed information about “Accounting - SQL DML”.

This block shows information about SQL DML (Data Manipulation Language) statements.

The following example shows both layouts, the report on the left, and the trace layout on the right.

**Accounting - SQL DML**

The field labels shown in the following sample layout of “Accounting - SQL DML” are described in the following section.

<table>
<thead>
<tr>
<th>Field</th>
<th>SQL DML</th>
<th>Average</th>
<th>Total</th>
<th>Trace</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECT</td>
<td>3.27</td>
<td>964</td>
<td>SELECT</td>
<td>488</td>
<td></td>
</tr>
<tr>
<td>INSERT</td>
<td>3.21</td>
<td>947</td>
<td>INSERT</td>
<td>37</td>
<td></td>
</tr>
<tr>
<td>ROWS</td>
<td>3.21</td>
<td>947</td>
<td>ROWS</td>
<td>36</td>
<td></td>
</tr>
<tr>
<td>UPDATE</td>
<td>3.83</td>
<td>1131</td>
<td>UPDATE</td>
<td>45</td>
<td></td>
</tr>
<tr>
<td>ROWS</td>
<td>5.03</td>
<td>1483</td>
<td>ROWS</td>
<td>36</td>
<td></td>
</tr>
<tr>
<td>MERGE</td>
<td>0.00</td>
<td>0</td>
<td>MERGE</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>DELETE</td>
<td>0.14</td>
<td>40</td>
<td>DELETE</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>ROWS</td>
<td>0.14</td>
<td>40</td>
<td>ROWS</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>DESCRIBE</td>
<td>2.79</td>
<td>824</td>
<td>DESCRIBE</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>DESC.TBL</td>
<td>0.00</td>
<td>0</td>
<td>DESC.TBL</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>PREPARE</td>
<td>3.01</td>
<td>888</td>
<td>PREPARE</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>OPEN</td>
<td>5.20</td>
<td>1533</td>
<td>OPEN</td>
<td>125</td>
<td></td>
</tr>
<tr>
<td>FETCH</td>
<td>3.36</td>
<td>991</td>
<td>FETCH</td>
<td>1331</td>
<td></td>
</tr>
<tr>
<td>ROWS</td>
<td>10.37</td>
<td>3060</td>
<td>ROWS</td>
<td>1000</td>
<td></td>
</tr>
<tr>
<td>CLOSE</td>
<td>3.36</td>
<td>990</td>
<td>CLOSE</td>
<td>109</td>
<td></td>
</tr>
<tr>
<td>DML-ALL</td>
<td>28.16</td>
<td>8308</td>
<td>DML-ALL</td>
<td>2136</td>
<td></td>
</tr>
</tbody>
</table>

**SELECT**

The number of SQL SELECT statements executed.

**Field Name:** QXSELECT

**INSERT**

The number of INSERT statements executed.

**Field Name:** QXINSRT

**INSERT - ROWS**

The number of rows inserted (DB2 field: QXRWSINSRTD).

**Field Name:** ARWINSRT

**UPDATE**

The number of UPDATE statements executed.

**Field Name:** QXUPDTE

**UPDATE - ROWS**

The number of rows updated (DB2 field: QXRWSUPDTD).

**Field Name:** ARWUPDAT

**MERGE**
SQL DML

The number of times a MERGE statement was executed.

Field Name: QXMERGE

DELETE

The number of DELETE statements executed.

Field Name: QXDELET

DELETE - ROWS

The number of rows deleted (DB2 field: QXRWSDELETD).

Field Name: ARWDELET

DESCRIBE

The number of DESCRIBE, DESCRIBE CURSOR, DESCRIBE INPUT, and DESCRIBE PROCEDURE statements executed. This number at the server location might not match the user application because of DDF's internal processing.

Field Name: QXDESC

DESC.TBL

The number of DESCRIBE TABLE statements executed.

Field Name: QXDSCRTB

PREPARE

The number of SQL PREPARE statements executed. This number at the server location might not match the user application because of DDF's internal processing.

Field Name: QXPREP

OPEN

The number of OPEN statements executed.

Field Name: QXOPEN

FETCH

The number of FETCH statements executed. This number at the server location might not match the user application because of DDF's internal processing.

Field Name: QXFETCH

FETCH - ROWS

The number of rows fetched (DB2 field: QXRWSFETCHD).

Field Name: ARWFETCH

CLOSE

The number of CLOSE statements executed. This number at the server location might not match the user application because of DDF's internal processing.

Field Name: QXCLOSE

DML-ALL

The total number of SQL DML statements executed.
Field Name: ASCDML
Termination - Abnormal

This topic shows detailed information about "Accounting - Termination - Abnormal".

This block shows a report for abnormal termination.

**Accounting - Termination - Abnormal**

The field labels shown in the following sample layout of "Accounting - Termination - Abnormal" are described in the following section.

<table>
<thead>
<tr>
<th>ABNORMAL TERM.</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPL.PROGR. ABEND</td>
<td>0</td>
</tr>
<tr>
<td>END OF MEMORY</td>
<td>0</td>
</tr>
<tr>
<td>RESOL.IN DOUBT</td>
<td>0</td>
</tr>
<tr>
<td>CANCEL FORCE</td>
<td>0</td>
</tr>
</tbody>
</table>

**APPL.PROGR. ABEND**

The number of abnormal terminations due to an application program abend.

Field Name: ASAPAPAB

**END OF MEMORY**

The number of abnormal terminations due to an end of memory. For example, accounting was invoked for an agent that was executing in an address space that experienced an abnormal end of memory.

Field Name: ASATENDM

**RESOL.IN DOUBT**

The number of abnormal terminations due to a resolve indoubt. For example, the recovery manager issued recover indoubt for a dependent thread that had not yet gone through end-of-task processing.

Field Name: ASATRIND

**CANCEL FORCE**

The number of abnormal terminations due to a stop force. For example, accounting was invoked for an agent that was executing when a -STOP DB2 MODE(FORCE) command was issued.

Field Name: ASATCANF
Termination - In Doubt

Termination - In Doubt
This topic shows detailed information about “Accounting - Termination - In Doubt”.

This block shows a report for in-doubt termination.

Accounting - Termination - In Doubt
The field labels shown in the following sample layout of “Accounting - Termination - In Doubt” are described in the following section.

<table>
<thead>
<tr>
<th>IN DOUBT</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPL.PGM ABEND</td>
<td>0</td>
</tr>
<tr>
<td>END OF MEMORY</td>
<td>0</td>
</tr>
<tr>
<td>END OF TASK</td>
<td>0</td>
</tr>
<tr>
<td>CANCEL FORCE</td>
<td>0</td>
</tr>
</tbody>
</table>

APPL.PGM ABEND
The number of work units indoubt due to an application program abend. The agent was indoubt when it abended.

Field Name: ASIDAPAB

END OF MEMORY
The number of work units indoubt due to an end of memory. For example, accounting was invoked for an agent that was indoubt when the address space in which it was executing experienced an abnormal end of memory.

Field Name: ASIDENDM

END OF TASK
The number of work units indoubt due to an end of task.

Field Name: ASIDENDT

CANCEL FORCE
The number of work units indoubt due to a stop force. For example, accounting was invoked for an agent that was indoubt when a -STOP DB2 MODE(FORCE) command was issued.

Field Name: ASIDCANF
**Termination - Normal**

This topic shows detailed information about “Accounting - Termination - Normal”.

This block shows a report for normal termination.

**Accounting - Termination - Normal**

The field labels shown in the following sample layout of “Accounting - Termination - Normal” are described in the following section.

<table>
<thead>
<tr>
<th>NORMAL TERM</th>
<th>AVERAGE</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>NEW USER</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>DEALLOCATION</td>
<td>1.00</td>
<td>1</td>
</tr>
<tr>
<td>APPL.PROGR. END</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>RESIGNON</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>DBAT INACTIVE</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>TYPE2 INACTIVE</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>RRS COMMIT</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>END USER THRESH</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>BLOCK STOR THR</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>STALENESS THR</td>
<td>0.00</td>
<td>0</td>
</tr>
</tbody>
</table>

**NEW USER**

The number of normal terminations due to a new user: either the authorization ID changed or there was a signon with the same authorization ID (normal).

*Field Name:* ASNTNEWU

**DEALLOCATION**

The number of normal terminations due to deallocation, which is a normal program termination.

*Field Name:* ASNTDEAL

**APPL.PROGR. END**

The number of normal terminations due to an application program end: the application program terminated without using DB2 protocols to end its connection to DB2. The agent did not abend so it is considered a normal termination.

*Field Name:* ASNTAPEN

**RESIGNON**

The number of normal terminations due to a resignon.

*Field Name:* ASNTRESI

**DBAT INACTIVE**

The number of normal terminations due to a DBAT becoming inactive.

*Field Name:* ASNTDBAT

**TYPE2 INACTIVE**

The number of times a DDF type 2 thread became inactive.

*Field Name:* ASNTDBA2

**RRS COMMIT**
Termination - Normal

The number of times a DB2 application using the RRS attach facility with accounting interval specified as COMMIT successfully committed a logical unit of work.

Field Name: ASRRSCOM

END USER THRESH

The number of times the threshold was reached for number of end user occurrences when data was accumulated by end user for DDF or RRSAF.

Field Name: ASNTTHEU

BLOCK STOR THR

The number of times the DB2 storage threshold for Accounting blocks was reached for data accumulated by end user for DDF or RRSAF.

Field Name: ASNTTHST

STALENESS THR

The number of times the threshold for the staleness was exceeded when data was accumulated by end user for DDF or RRSAF.

Field Name: ASNTTHSL
Times - Class 1 - Application Time

This topic shows detailed information about “Accounting - Times - Class 1 - Application Time”.

This block shows information for the Application Time class 1.

The following example shows both layouts, the report on the left, and the trace layout on the right.

Accounting - Times - Class 1 - Application Time

The field labels shown in the following sample layout of “Accounting - Times - Class 1 - Application Time” are described in the following section.

Report:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>AVERAGE</td>
<td>21.452714</td>
<td>TIMES/EVENTS</td>
<td>6:37.60226</td>
</tr>
<tr>
<td>ELAPSED TIME</td>
<td>21.452714</td>
<td>NONNESTED</td>
<td>21.452714</td>
</tr>
<tr>
<td>STORED PROC</td>
<td>0.000000</td>
<td>STORED PROC</td>
<td>0.000000</td>
</tr>
<tr>
<td>UDF</td>
<td>0.000000</td>
<td>UDF</td>
<td>0.000000</td>
</tr>
<tr>
<td>TRIGGER</td>
<td>0.000000</td>
<td>TRIGGER</td>
<td>0.000000</td>
</tr>
<tr>
<td>CP CPU TIME</td>
<td>7.217101</td>
<td>CP CPU TIME</td>
<td>7.217101</td>
</tr>
<tr>
<td>AGENT</td>
<td>7.172907</td>
<td>AGENT</td>
<td>7.172907</td>
</tr>
<tr>
<td>STORED PROC</td>
<td>0.000000</td>
<td>STORED PROC</td>
<td>0.000000</td>
</tr>
<tr>
<td>UDF</td>
<td>0.000000</td>
<td>UDF</td>
<td>0.000000</td>
</tr>
<tr>
<td>TRIGGER</td>
<td>0.000000</td>
<td>TRIGGER</td>
<td>0.000000</td>
</tr>
<tr>
<td>PAR.TASKS</td>
<td>0.044194</td>
<td>PAR.TASKS</td>
<td>0.238651</td>
</tr>
<tr>
<td>SECP CPU</td>
<td>0.000000</td>
<td>SECP CPU</td>
<td>0.000000</td>
</tr>
<tr>
<td>SE CPU TIME</td>
<td>4.034602</td>
<td>SE CPU TIME</td>
<td>4.034602</td>
</tr>
<tr>
<td>NONNESTED</td>
<td>4.034602</td>
<td>NONNESTED</td>
<td>4.034602</td>
</tr>
<tr>
<td>STORED PROC</td>
<td>0.000000</td>
<td>STORED PROC</td>
<td>0.000000</td>
</tr>
<tr>
<td>UDF</td>
<td>0.000000</td>
<td>UDF</td>
<td>0.000000</td>
</tr>
<tr>
<td>TRIGGER</td>
<td>0.000000</td>
<td>TRIGGER</td>
<td>0.000000</td>
</tr>
<tr>
<td>PAR.TASKS</td>
<td>0.000000</td>
<td>PAR.TASKS</td>
<td>0.000000</td>
</tr>
<tr>
<td>SUSPEND TIME</td>
<td>0.000000</td>
<td>SUSPEND TIME</td>
<td>0.000000</td>
</tr>
<tr>
<td>AGENT</td>
<td>N/A</td>
<td>AGENT</td>
<td>N/A</td>
</tr>
<tr>
<td>PAR.TASKS</td>
<td>N/A</td>
<td>PAR.TASKS</td>
<td>N/A</td>
</tr>
<tr>
<td>STORED PROC</td>
<td>0.000000</td>
<td>STORED PROC</td>
<td>0.000000</td>
</tr>
<tr>
<td>UDF</td>
<td>0.000000</td>
<td>UDF</td>
<td>0.000000</td>
</tr>
</tbody>
</table>

ELAPSED TIME

The class 1 elapsed time of the allied agent.

Special Considerations:

1. If the begin time equals zero, or if the end time minus begin time equals zero or is negative, N/C is shown.
2. Threads that can be reused, such as CICS protected threads or IMS/VS wait-for-input message regions, can include time during which the thread was inactive and waiting for work.
3. Elapsed time to process distributed requests is included for allied-distributed threads.
4. This time includes the time for processing SQL statements issued by stored procedures, user-defined functions, or triggers.
Times - Class 1 - Application Time

5. In query CP, sysplex query, or utility parallelism, this is the time shown in the originating record, which overlaps the elapsed times shown in the parallel records.

Field Name: ADRECETT
This is an exception field.

ELAPSED TIME - NONNESTED
The class 1 elapsed time of the allied agent.

Special Considerations:
1. If the begin time equals zero, or if the end time minus begin time equals zero or is negative, N/C is shown.
2. Threads that can be reused, such as CICS protected threads or IMS/VS wait-for-input message regions, can include time during which the thread was inactive and waiting for work.
3. Elapsed time to process distributed requests is included for allied-distributed threads.
4. This time includes the time for processing SQL statements issued by stored procedures, user-defined functions, or triggers.
5. In query CP, sysplex query, or utility parallelism, this is the time shown in the originating record, which overlaps the elapsed times shown in the parallel records.

Field Name: ADNNNET1

ELAPSED TIME - STORED PROC
An accumulated and consumed time for stored procedures. It consists of the following parts:

• The total elapsed time spent by the allied agent in stored procedures. A stored procedure may initiate a trigger or invoke a user-defined function. The time spent for initiation or invocation is not included in this counter.
• Accumulated elapsed time consumed when running stored procedure requests on the main application execution unit. As these stored procedures run entirely within DB2, this time represents class 1 and class 2 time.

Field Name: ADELTSP1

ELAPSED TIME - UDF
The total elapsed time spent in user-defined function (UDF) requests processed in a DB2 stored procedure or WLM address space. Non-inline UDF times are included in this time if the native UDF was called on a nested task and was not invoked by the main application execution unit.

This time includes times executing SQL and times consumed executing user-defined functions on the main application execution unit.

A user-defined function may initiate a trigger or invoke a stored procedure. The time spent is not included in this counter.

Field Name: AWAELUD1

ELAPSED TIME - TRIGGER
The total elapsed time spent by the allied agent in triggers.
**Times - Class 1 - Application Time**

A trigger may invoke a stored procedure or a user-defined function. The time spent there is not included in this counter.

For triggers there is no distinction between class 1 and class 2 CPU time: all processing controlled by a trigger is within DB2.

**Note:** This field is not normally shown in the short layouts but can be included with UTR.

**Field Name:** ADTRET

**CP CPU TIME**

The class 1 CPU time in an application. It indicates:

- The class 1 CPU time of the allied agent, which may include the accumulated class 1 TCB time for processing stored procedures, user-defined functions, and triggers.
- The accumulated CPU time for processing parallel tasks. This is valid for query CP parallelism, sysplex query parallelism, and parallel tasks generated by utilities.
- In sysplex query parallelism, the individual CPU times are normalized by the conversion factor of the parallel tasks that is related to the originating task. In sysplex query parallelism, only CPU times of parallel tasks, running on the same member of the SYSPLEX group as the originating task, are included.

This CPU time does not include time that is consumed on an IBM specialty engine.

**Field Name:** ADCPUT

This is an exception field.

**CP CPU TIME - AGENT**

It comprises the class 1 CPU time of the allied agent, which may include the accumulated class 1 CPU time for processing stored procedures, user-defined functions, and triggers if present.

CPU time for processing parallel tasks is not charged to this counter.

This CPU time does not include the CPU time that is consumed on an IBM specialty engine.

**Field Name:** ADAGENT1

This is an exception field.

**CP CPU TIME - AGENT - NONNESTED**

The class 1 CPU time of the nonnested activity of the allied agent.

**Field Name:** ADNNEST1

**CP CPU TIME - AGENT - STORED PRC**

An accumulated and consumed time for stored procedures that consists of following parts:

- The CPU time accumulated in DB2 for processing SQL CALL statements in the stored procedures or WLM address space. This time is only calculated if accounting class 1 is active.
Times - Class 1 - Application Time

- The accumulated CPU time consumed when running stored procedure requests on the main application execution unit. This time does not include CPU time consumed on an IBM specialty engine. As these stored procedures run entirely within DB2, this time represents class 1 and class 2 time.

**Field Name:** ADCPUSP1

**CP CPU TIME - AGENT - UDF**

The accumulated CPU time consumed executing user-defined functions. This time does not include CPU consumed on an IBM specialty engine. It consists of following parts:

- The accumulated CPU time used to satisfy UDF requests processed in a DB2 stored procedure or WLM address space. This time is only calculated if accounting class 1 is active.
- The accumulated CPU time consumed executing user-defined functions on the main application execution unit. This time represents class 1 and class 2 time, because these UDFs run entirely within DB2,

**Prior to DB2 10:** The accumulated CPU time used to satisfy UDF requests processed in a DB2 stored procedure or WLM address space. This time is only calculated if accounting class 1 is active. This time does not include the CPU time consumed on an IBM specialty engine.

**Field Name:** AWACPUD1

**CP CPU TIME - AGENT - TRIGGER**

The accumulated CPU time consumed while executing under the control of triggers.

For triggers there is no distinction between class 1 and class 2 CPU time. All processing controlled by a trigger is within DB2.

**Field Name:** ADTRCP

**CP CPU TIME - PAR.TASKS**

The sum of the CPU times of the parallel tasks running in an application. It can also include the accumulated class 1 CPU time for processing stored procedures, user-defined functions, and triggers if present.

These parallel tasks can be query CP, sysplex query parallel tasks, or parallel tasks produced by utilities. In sysplex query parallelism, the accumulated time reflects only parallel tasks running on the same DB2 subsystem as the originating task.

This time does not include CPU time consumed on an IBM specialty engine.

CPU time for agent tasks is not charged to this counter.

**Field Name:** ADCPUPLL

**SECP CPU**

The accumulated CPU time that ran on a standard CP for work eligible on an IBM specialty engine.

For records for the parent tasks in parallel queries, this value reflects zIIP-eligible time for the parent and the child tasks. Child task records have a value of 0.

**Field Name:** AWACZEL
SE CPU TIME

The sum of several accumulated CPU times consumed while running on an IBM specialty engine in all environments. These times are consumed when:

- Running stored procedure requests and triggers on the main application execution unit.
- Satisfying stored procedure requests processed in a DB2 stored procedure or WLM address space. SQLP times are included in this time if the SQLP was called on a nested task and was not invoked by the main application execution unit.
- Satisfying UDF requests processed in a DB2 stored procedure or WLM address space.
- Running triggers on a nested task.
- Running parallel tasks in an application which contains the accumulated CPU time used to satisfy UDF requests.

Note: All CPU times of an IBM Specialty Engine (SE) that are reported in DB2 trace records are already normalized by DB2 to the speed of the purpose processor.

Field Name: AWACC1Z

SE CPU TIME - NONNESTED

The class 1 CPU time for nonnested activity on the main application task consumed while running on an IBM specialty engine. This time ignores the CPU time that is consumed when running stored procedure requests, UDF requests, or any triggers on the main application execution unit on an IBM specialty engine.

Field Name: ADSENNC1

SE CPU TIME - STORED PROC

An accumulated and consumed time for stored procedures on an IBM specialty engine that consists of following parts:

- The time processed in a DB2 stored procedure or WLM address space. SQLP times are included in this time if the SQLP was called on a nested task and was not invoked by the main application execution unit.
- The time when running on the main application execution unit. This part of time counts for class 1 and class 2 time, because these stored procedures run entirely within DB2.

Field Name: ADSESP1

SE CPU TIME - UDF

An accumulated and consumed time for user-defined functions on an IBM specialty engine that consists of following parts:

- The accumulated CPU time used to satisfy UDF requests processed in a DB2 stored procedure or WLM address space.
- The accumulated CPU time consumed when running user-defined functions on the main application execution unit. This time represents class 1 and class 2 time, because these UDFs run entirely within DB2.

For versions prior to DB2 10: The accumulated CPU time that is required for UDF requests processed in a DB2 stored procedure or WLM address space while running on an IBM specialty engine.
Field Name: AWACSEU1

SE CPU TIME - TRIGGER

The accumulated CPU time consumed on an IBM specialty engine while running triggers on a nested task or on the main application execution unit.

Field Name: AWACTRZ

SE CPU TIME - PAR.TASKS

The sum of the CPU times of the parallel tasks, or roll-up autonomous tasks that are running in an application on an IBM specialty engine.

It contains the accumulated CPU time that is used to satisfy UDF requests, which are processed in a DB2 stored procedure or WLM address space while running on an IBM specialty engine.

Field Name: ADSEPLL1

SUSPEND TIME

The amount of application suspension time spent outside DB2.

Field Name: ASUSTCL1

SUSPEND TIME - STORED PROC

The total elapsed waiting time for an available TCB before the stored procedure could be scheduled.

Field Name: QWACCAST

SUSPEND TIME - UDF

The total elapsed time spent waiting for an available TCB before the user-defined function could be scheduled.

Field Name: QWACUDST
Times - Class 1 - Elapsed Time Distribution

This topic shows detailed information about "Accounting - Times - Class 1 - Elapsed Time Distribution".

The elapsed time distribution block shows the distribution of the task. For threads exploiting parallelism, only the nonparallel part is taken into account.

The following example applies to both, the report layout and the trace layout.

**Accounting - Times - Class 1 - Elapsed Time Distribution**

The field labels shown in the following sample layout of "Accounting - Times - Class 1 - Elapsed Time Distribution" are described in the following section.

<table>
<thead>
<tr>
<th>ELAPSED TIME DISTRIBUTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPL</td>
</tr>
<tr>
<td>DB2</td>
</tr>
<tr>
<td>SUSP</td>
</tr>
</tbody>
</table>

**APPL**

The ratio of the elapsed application time, expressed as a percentage of the total elapsed time.

**Field Name:** ARATAPL

**DB2**

The ratio of the elapsed DB2 time, expressed as a percentage of the total elapsed time.

**Field Name:** ARATDB2

**SUSP**

The ratio of the DB2 suspension time, expressed as a percentage of the total elapsed time.

**Field Name:** ARATSUS
### Times - Class 2 - DB2 Time

**Times - Class 2 - DB2 Time**

This topic shows detailed information about “Accounting - Times - Class 2 - DB2 Time”.

This block shows information for DB2 class 2.

The following example shows both layouts, the report on the left, and the trace layout on the right.

**Accounting - Times - Class 2 - DB2 Time**

The field labels shown in the following sample layout of “Accounting - Times - Class 2 - DB2 Time” are described in the following section.

**Report:**

<table>
<thead>
<tr>
<th>Field</th>
<th>Report Value</th>
<th>Trace Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ELAPSED TIME</td>
<td>21.196309</td>
<td>6:23.48603</td>
</tr>
<tr>
<td>NONNESTED</td>
<td>21.196309</td>
<td>6:23.48603</td>
</tr>
<tr>
<td>STORED PROC</td>
<td>0.000000</td>
<td>0.000000</td>
</tr>
<tr>
<td>UDF</td>
<td>0.000000</td>
<td>0.000000</td>
</tr>
<tr>
<td>TRIGGER</td>
<td>0.000000</td>
<td>0.000000</td>
</tr>
<tr>
<td>CP CPU TIME</td>
<td>7.134566</td>
<td>2:27.81217</td>
</tr>
<tr>
<td>AGENT</td>
<td>7.134299</td>
<td>2:27.81191</td>
</tr>
<tr>
<td>NONNESTED</td>
<td>7.134299</td>
<td>2:27.81191</td>
</tr>
<tr>
<td>STORED PROC</td>
<td>0.000000</td>
<td>0.000000</td>
</tr>
<tr>
<td>UDF</td>
<td>0.000000</td>
<td>0.000000</td>
</tr>
<tr>
<td>TRIGGER</td>
<td>0.000000</td>
<td>0.000000</td>
</tr>
<tr>
<td>PAR.TASKS</td>
<td>0.000267</td>
<td>0.000259</td>
</tr>
<tr>
<td>SECP CPU</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>SE CPU TIME</td>
<td>4.034602</td>
<td>19.274049</td>
</tr>
<tr>
<td>NONNESTED</td>
<td>4.034602</td>
<td>19.274049</td>
</tr>
<tr>
<td>STORED PROC</td>
<td>0.000000</td>
<td>0.000000</td>
</tr>
<tr>
<td>UDF</td>
<td>0.000000</td>
<td>0.000000</td>
</tr>
<tr>
<td>TRIGGER</td>
<td>0.000000</td>
<td>0.000000</td>
</tr>
<tr>
<td>PAR.TASKS</td>
<td>0.000000</td>
<td>0.000000</td>
</tr>
<tr>
<td>SUSPEND TIME</td>
<td>8.990994</td>
<td>3:34.39482</td>
</tr>
<tr>
<td>AGENT</td>
<td>8.990994</td>
<td>3:34.39482</td>
</tr>
<tr>
<td>PAR.TASKS</td>
<td>0.000000</td>
<td>0.000000</td>
</tr>
<tr>
<td>STORED PROC</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>UDF</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>NOT ACCOUNT.</td>
<td>1.126713</td>
<td>2.005250</td>
</tr>
<tr>
<td>DB2 ENT/EXIT</td>
<td>12.00</td>
<td>19.968</td>
</tr>
<tr>
<td>EN/EX-STPROC</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td>EN/EX-UDF</td>
<td>0.00</td>
<td>0</td>
</tr>
</tbody>
</table>

**ELAPSED TIME**

The class 2 elapsed time of the allied agent accumulated in DB2.

**Field Name:** ADDB2ETT

This is an exception field.

**ELAPSED TIME - NONNESTED**

The class 2 elapsed time for nonnested activity accumulated in DB2 for the allied agent. This time does not include the time spent in DB2 processing SQL statements issued by stored procedures, user-defined functions, or triggers.

**Special Considerations**

- The time for most thread allocation and certain abend conditions is not reflected in this time.
- The elapsed time for distributed processing is included in the elapsed time of allied-distributed threads.
Times - Class 2 - DB2 Time

- In query CP, sysplex query, or utility parallelism, this is the time shown in the originating record, which overlaps the elapsed times shown in the parallel records.

**Note:** This field is not normally shown in the short layouts but can be included with UTR.

**Field Name:** QWACASC

**ELAPSED TIME - STORED PROC**
An accumulated and consumed time for stored procedures that consists of following parts:
- The total elapsed time that the allied agent spent when running SQL in the stored procedures or WLM address space. A stored procedure may initiate a trigger or invoke a user-defined function. This time is not included in this counter.
- Accumulated elapsed time consumed when running stored procedure requests on the main application execution unit. As these stored procedures run entirely within DB2, this time represents class 1 and class 2 time.

**Field Name:** ADELTSP2

**ELAPSED TIME - UDF**
The total elapsed time that is spent executing SQL using user-defined function (UDF) requests that are processed in a DB2 stored procedure or WLM address space. This time includes time required to connect and disconnect the UDF task. Non-inline UDF times are included in this time if the native UDF was called on a nested task and was not invoked by the main application execution unit.

This time includes the elapsed time that is consumed when executing user-defined functions on the main application execution unit.

A user-defined function may initiate a trigger or invoke a stored procedure. Any time spent there is not included in this counter.

**Field Name:** AWAELUD2

**ELAPSED TIME - TRIGGER**
The total elapsed time spent by the allied agent in triggers.

A trigger may invoke a stored procedure or a user-defined function. The time spent there is not included in this counter.

For triggers there is no distinction between class 1 and class 2 CPU time: all processing controlled by a trigger is within DB2.

**Note:** This field is not normally shown in the short layouts but can be included with UTR.

**Field Name:** ADTRET

**CP CPU TIME**
The class 2 CPU time (in DB2). It indicates:
- The class 2 CPU time for the allied agent. This includes the accumulated class 2 TCB time for processing any stored procedures, user-defined functions, and triggers.
**Times - Class 2 - DB2 Time**

- The accumulated CPU time for processing parallel tasks. This is valid for query CP parallelism, sysplex query parallelism, and parallel tasks generated by utilities.

- For batch reporting, in sysplex query parallelism, the individual CPU times are normalized by the conversion factor of the parallel tasks, related to the originating task.

For online monitoring, in sysplex query parallelism, only CPU times of parallel tasks, running on the same member of the sysplex group as the originating task, are included.

This CPU time does not include time that is consumed on an IBM specialty engine.

**Field Name:** ADDBCPUT

This is an *exception* field.

**CP CPU TIME - AGENT**

It comprises the class 2 CPU time of the allied agent. This time includes the accumulated class 2 CPU time for processing stored procedures, user-defined functions, and triggers, if present. CPU time for processing parallel tasks is not charged to this counter.

This CPU time does not include the CPU time that is consumed on an IBM specialty engine.

**Field Name:** ADAGENT2

**CP CPU TIME - AGENT - NONNESTED**

The class 2 CPU time of the nonnested activity of the allied agent.

This value indicates the CPU time the allied agent spent in DB2 for nonnested activity. This time does not include the time for processing SQL statements issued by stored procedures, user-defined functions, or triggers.

**Special Considerations:**

1. For allied-distributed threads, this does not include the time used to process distributed SQL. For DBAT-distributed threads, this includes only processing at this location.

2. Most thread allocation and certain abend conditions are not included.

3. This time does not include the time for processing parallel tasks generated by utilities or in query CP or sysplex query parallelism.

**Field Name:** ADNNEST2

**CP CPU TIME - AGENT - STORED PRC**

An accumulated and consumed time for stored procedures that consists of the following information:

- The CPU time accumulated in DB2 for processing SQL statements issued by stored procedures processed in a DB2 stored procedure or WLM address space. This time is only calculated if accounting class 2 is active.

- In DB2 time needed to connect and disconnect the SP task for non-SQLP stored procedures.

- SQLP times are included in this time if the SQLP was called on a nested task and was not invoked by the main application execution unit.

- CPU time that is consumed when running stored procedure requests on the main application execution unit.
This time does not include the CPU time consumed on an IBM specialty engine.

Field Name: ADCPUSP2

CP CPU TIME - AGENT - UDF

The accumulated CPU time consumed executing user-defined functions. This time does not include CPU time consumed on an IBM specialty engine. It consists of following parts:

- The accumulated CPU time consumed in DB2 when processing SQL statements that were issued by UDFs in a DB2 stored procedure or WLM address space. This time also includes the DB2 time required to connect or disconnect the UDF task. It is only calculated if accounting class 2 is active.

- The accumulated CPU time consumed executing user-defined functions on the main application execution unit. This time represents class 1 and class 2 time, because these UDFs run entirely within DB2.

For versions prior to DB2 10: The accumulated CPU time consumed in DB2 when processing SQL statements that were issued by UDFs in a DB2 stored procedure or WLM address space. This time also includes the DB2 time required to connect and disconnect the UDF task. This time does not include CPU time consumed on an IBM specialty engine. It is only calculated if Accounting class 2 is active.

Field Name: AWACPUUD2

CP CPU TIME - AGENT - TRIGGER

The accumulated CPU time consumed while executing under the control of triggers.

For triggers there is no distinction between class 1 and class 2 CPU time. All processing controlled by a trigger is within DB2.

Field Name: ADTRCP

CP CPU TIME - PAR.TASKS

The sum of the CPU times of the parallel tasks running in DB2. These tasks can be query CP, sysplex query parallel tasks, parallel tasks produced by utilities, or rollup autonomous tasks.

In sysplex query parallelism, the accumulated time reflects only parallel tasks running on the same DB2 subsystem as the originating task.

This time does not include the CPU time consumed on an IBM specialty engine.

Field Name: ADDBCPC2

SE CPU TIME

The accumulated and consumed class 2 time on an IBM specialty engine that consists of times for nonnested, stored procedures, user-defined functions, triggers, and parallel tasks.

Note: All CPU times of an IBM Specialty Engine (SE) that are reported in DB2 trace records are already normalized by DB2 to the speed of the purpose processor.

Field Name: AWACC2Z
**Times - Class 2 - DB2 Time**

**SE CPU TIME - NONNESTED**

The class 2 CPU time for nonnested activity on the main application task consumed while running on an IBM specialty engine.

*Field Name:* ADSENNC2

**SE CPU TIME - STORED PROC**

An accumulated and consumed time for stored procedures on an IBM specialty engine that consists of following parts:

- The time consumed in DB2 in a DB2 stored procedure or WLM address space. SQLP times are included in this time if the SQLP was called on a nested task and was not invoked by the main application execution unit.
- The time when running on the main application execution unit. This part of time counts for class 1 and class 2 time, because these stored procedures run entirely within DB2.

This time is a subset of QWACSP_CLS1SE.

*Field Name:* ADSESP2

**SE CPU TIME - UDF**

The accumulated and consumed time for user-defined functions on an IBM specialty engine that consists of following parts:

- The accumulated CPU time consumed in DB2 processing SQL statements issued by UDFs processed in a DB2 stored procedure or WLM address space. This time is a subset of QWACUDF_CLS1SE.
- The accumulated CPU time that is consumed when running user-defined functions on the main application execution unit. This time represents class 1 and class 2 time, because these UDFs run entirely within DB2.

*For DB2 9:* The accumulated CPU time that is consumed in DB2 processing SQL statements issued by UDFs processed in a DB2 stored procedure or WLM address space while running on an IBM specialty engine.

*Field Name:* AWACSEU2

**SE CPU TIME - TRIGGER**

The accumulated CPU time consumed on an IBM specialty engine while running triggers on a nested task or on the main application execution unit.

*Field Name:* AWACTRZ

**SE CPU TIME - PAR.TASKS**

The sum of the CPU times of the parallel tasks, or roll-up autonomous task that are running in DB2 on an IBM specialty engine.

It contains the accumulated CPU time that is consumed in DB2 when processing SQL statements that are issued by UDFs processed in a DB2 stored procedure or WLM address space while running on an IBM specialty engine.

*Field Name:* ADSEPLL2

**SUSPEND TIME**

The waiting time for all types of class 3 suspensions by the originating task and parallel tasks, if parallelism is employed.

*Field Name:* ADTSUST
**Times - Class 2 - DB2 Time**

This is an exception field.

**SUSPEND TIME - AGENT**

The waiting time of the allied agent for all types of class 3 suspension
This counts class 3 suspension time within nested activity.
Suspension time of parallel tasks in query or utility parallelism is not included.

**Field Name:** ADTCBCL3

This is an exception field.

**SUSPEND TIME - PAR.TASKS**

The sum of the suspension times spent for parallel tasks. These tasks can be query CP or sysplex query parallel tasks, parallel tasks produced by utilities, or roll-up autonomous tasks.

**Field Name:** ADCPCL3T

**NOT ACCOUNT.**

The time not accounted in DB2. This time determines whether there is a large percentage of time that has not been captured within the DB2 accounting record and whether system monitoring tools (such as RMF™) should be examined to determine the cause of a performance problem.

In query or utility parallelism, it is the unaccounted time of the originating task only.

Check the DB2 accounting class 2 elapsed time that is not recorded as class 2 CPU time or class 3 suspensions. The following list shows why DB2 Class 2 Not Accounted time can be significant:

- Too much detailed online tracing, or problems with vendor performance monitors. This situation is usually the primary cause of high not-accounted-for time on systems that are not CPU-constrained.
- Running in a very high CPU utilization environment and waiting for CPU cycles if DB2 work WLM service class goals are not set properly.
- Running in a high MVS paging environment and waiting for storage allocation.
- The IBM specialty engines are highly utilized and the SYS1.PARMLIB(IEAOPTxx) member has the following settings: IIPHONORPRIORITY=NO and IFAHONORPRIORITY=NO.
- Frequent gathering of data set statistics (SMF 46 Type 2 records) DD consolidation overhead (z/OS parm DDCONS=YES DETAIL).
- CF Lock Structure system managed DUXPLEXing since DB2 is not informed about related suspensions waits.
- In very I/O intensive environments, the Media Manager might be running out of request blocks.
- Time spent waiting for parallel tasks to complete (when query parallelism is used for the query).
- HSM (Hierarchical Storage Manager) data set recall is an asynchronous process.
- Waiting for requests to be returned from SNA DB2 Server.
- Data set open contention related to PCLOSET being too small.
Times - Class 2 - DB2 Time

- DB2 internal suspend and resume looping when several threads are waiting for the same resource.
- For DDF requesters, this value can be very large because it includes the time the requesting thread waited for responses from the server. Because there can be asynchronous activity at the requester, the DDF time is only an approximation.
  Refer to the DDF server requester elapsed time to determine the amount of time the thread waited for server responses.

Field Name: ADNOTACC

DB2 ENT/EXIT

The total number of DB2 entry and exit events processed by the allied address space to calculate the elapsed time in DB2 and the processor time.
This counter does not include the SQL entry and exit events processed by stored procedures.

Field Name: QWACARNA
This is an exception field.

EN/EX-STPROC

The number of SQL entry or exit events performed by stored procedures.
This number is only calculated if accounting class 2 is active.

Field Name: QWACSPNE

EN/EX-UDF

The number of SQL entry/exit events performed by user-defined functions.
This is only calculated if accounting class 2 is active.

Field Name: QWACUDNE
Times - Class 2 - Time Distribution

This topic shows detailed information about “Accounting - Times - Class 2 - Time Distribution”.

The class 2 time distribution block shows the distribution of the active-in-DB2 time, the not-accounted time, and the suspension time, of the originating task. For threads exploiting query parallelism, only the nonparallel part is taken into account.

The following example applies to both, the report layout and the trace layout.

**Accounting - Times - Class 2 - Time Distribution**

The field labels shown in the following sample layout of “Accounting - Times - Class 2 - Time Distribution” are described in the following section.

```
CLASS 2 TIME DISTRIBUTION

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>The ratio of the agent DB2 CPU time, expressed as a percentage of the DB2 elapsed time.</td>
</tr>
<tr>
<td>SECPU</td>
<td>The ratio of the agent DB2 SE (IBM specialty engine) CPU time, expressed as a percentage of the DB2 elapsed time.</td>
</tr>
<tr>
<td>NOTACC</td>
<td>The ratio of the DB2 not accounted time, expressed as a percentage of the DB2 elapsed time.</td>
</tr>
<tr>
<td>SUSP</td>
<td>The ratio of the agent DB2 suspension time, expressed as a percentage of the DB2 elapsed time.</td>
</tr>
</tbody>
</table>
```

Field Name: ARATCPU

Field Name: ARATCSE

Field Name: ARATNAC

Field Name: ARATSUP
Times - Class 3 - Suspensions

This topic shows detailed information about “Accounting - Times - Class 3 - Suspensions”.

This block shows information for Class 3 Suspensions.

The following example shows both layouts, the report on the left, and the trace layout on the right.

Accounting - Times - Class 3 - Suspensions

The field labels shown in the following sample layout of “Accounting - Times - Class 3 - Suspensions” are described in the following section.

Report: Trace:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Class 3 Suspensions</th>
<th>Average Time</th>
<th>AV.EVENT</th>
<th>Class 3 Suspensions</th>
<th>Elapsed Time</th>
<th>EVENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOCK/LATCH(DB2+IRLM)</td>
<td>0.000000</td>
<td>0.00</td>
<td></td>
<td>LOCK/LATCH(DB2+IRLM)</td>
<td>31.755074</td>
<td>1</td>
</tr>
<tr>
<td>IRLM LOCK+LATCH</td>
<td>0.000000</td>
<td>0.00</td>
<td></td>
<td>IRLM LOCK+LATCH</td>
<td>31.755074</td>
<td>1</td>
</tr>
<tr>
<td>DB2 LATCH</td>
<td>0.000000</td>
<td>0.00</td>
<td></td>
<td>DB2 LATCH</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>DATABASE I/O</td>
<td>0.000000</td>
<td>0.00</td>
<td></td>
<td>DATABASE I/O</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>LOG WRITE I/O</td>
<td>0.000000</td>
<td>0.00</td>
<td></td>
<td>LOG WRITE I/O</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>OTHER READ I/O</td>
<td>0.000000</td>
<td>0.00</td>
<td></td>
<td>OTHER READ I/O</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>OTHER WRITE I/O</td>
<td>0.000000</td>
<td>0.00</td>
<td></td>
<td>OTHER WRITE I/O</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>SER.TASK SWITCH</td>
<td>0.007394</td>
<td>4.00</td>
<td></td>
<td>SER.TASK SWITCH</td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>UPDATE COMMIT</td>
<td>0.0000003</td>
<td>1.00</td>
<td></td>
<td>UPDATE COMMIT</td>
<td>0.0001111</td>
<td>1</td>
</tr>
<tr>
<td>OPEN/CLOSE</td>
<td>0.000000</td>
<td>0.00</td>
<td></td>
<td>OPEN/CLOSE</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>SYSLOGREC</td>
<td>0.000000</td>
<td>0.00</td>
<td></td>
<td>SYSLOGREC</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>EXT/DL/DEF</td>
<td>0.000000</td>
<td>0.00</td>
<td></td>
<td>EXT/DL/DEF</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>OTHER SERVICE</td>
<td>0.007330</td>
<td>3.00</td>
<td></td>
<td>OTHER SERVICE</td>
<td>0.002566</td>
<td>2</td>
</tr>
<tr>
<td>ARC.LOG(QUIES)</td>
<td>0.000000</td>
<td>0.00</td>
<td></td>
<td>ARC.LOG(QUIES)</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>LOG READ</td>
<td>0.000000</td>
<td>0.00</td>
<td></td>
<td>LOG READ</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>DRAIN Lock</td>
<td>0.000000</td>
<td>0.00</td>
<td></td>
<td>DRAIN Lock</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>CLAIM RELEASE</td>
<td>0.000000</td>
<td>0.00</td>
<td></td>
<td>CLAIM RELEASE</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>PAGE LATCH</td>
<td>0.000000</td>
<td>0.00</td>
<td></td>
<td>PAGE LATCH</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>NOTIFY MSGS</td>
<td>0.000000</td>
<td>0.00</td>
<td></td>
<td>NOTIFY MSGS</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>GLOBAL CONTENTION</td>
<td>0.000000</td>
<td>0.00</td>
<td></td>
<td>GLOBAL CONTENTION</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>COMMIT PH1 WRITE I/O</td>
<td>0.000000</td>
<td>0.00</td>
<td></td>
<td>COMMIT PH1 WRITE I/O</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>ASYNCH CF REQUESTS</td>
<td>0.000000</td>
<td>0.00</td>
<td></td>
<td>ASYNCH CF REQUESTS</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>TCP/IP LOB XML</td>
<td>0.000000</td>
<td>0.00</td>
<td></td>
<td>TCP/IP LOB XML</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>ACCELERATOR</td>
<td>0.000000</td>
<td>0.00</td>
<td></td>
<td>ACCELERATOR</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>AUTONOMOUS PROCEDURE</td>
<td>0.000000</td>
<td>0.00</td>
<td></td>
<td>AUTONOMOUS PROCEDURE</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>PS SYNCHRONIZATION</td>
<td>0.000000</td>
<td>0.00</td>
<td></td>
<td>PS SYNCHRONIZATION</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>TOTAL CLASS 3</td>
<td>0.007394</td>
<td>4.00</td>
<td></td>
<td>TOTAL CLASS 3</td>
<td>31.757751</td>
<td>4</td>
</tr>
</tbody>
</table>

LOCK/LATCH(DB2+IRLM) - AVERAGE TIME/ELAPSED TIME

This field depends on the DB2 version that is installed:

- **DB2 10 or later**: The sum of the accumulated wait time due to local contention for locks and the accumulated wait time due to latch contention.

- **Prior to DB2 10**: The accumulated lock and latch elapsed time.

**Note**: The internally defined field adjusts the original DB2 value. DB2 counts each event twice, one for the entry and one for the exit.

**Field Name**: AWTLOLA

LOCK/LATCH(DB2+IRLM) - AV.EVENT/EVENTS

This field depends on the DB2 version that is installed:

- **DB2 10 or later**: The sum of the number of wait trace events processed for waits for local contention for locks and the number of wait trace events processed for waits for latch contention.

- **Prior to DB2 10**: The number of wait trace events processed for lock/latch.
**Times - Class 3 - Suspensions**

**Note:** The internally defined field adjusts the original DB2 value. DB2 counts each event twice, one for the entry and one for the exit.

**Field Name:** ADLLSUSC

This is an *exception* field.

**IRLM LOCK+LATCH - AVERAGE TIME/ELAPSED TIME**

This field depends on the DB2 version that is installed:

- **DB2 10 or later:** The accumulated wait time because of local contention for locks. The term *local contention* is used to differentiate from *global contention* (which is reported in QWACAWTJ). Local contention does not require intersystem communication. The contention is detected and resolved entirely within this subsystem.

- **Prior to DB2 10:** The accumulated lock and latch elapsed time. It indicates the elapsed time the allied agent waited for locks and latches in DB2. This value does not include suspensions because of group-level lock contentions in a data sharing environment. When the event completes, the ending time is used to calculate the total elapsed wait time. The result is added to the previously saved lock and latch wait time in DB2. If class 9 is not active, the time shown does not include the time for internal latching.

**Field Name:** QWACAWT

This is an *exception* field.

**IRLM LOCK+LATCH - AV.EVENT/EVENTS**

The number of wait trace events processed for waits for local contention for locks.

**Note:** The internally defined field adjusts the original DB2 value. DB2 counts each event twice, one for the entry and one for the exit.

**Field Name:** ADLBSUSC

**DB2 LATCH - AVERAGE TIME/ELAPSED TIME**

The accumulated wait time because of latch contention.

**Field Name:** QWACAWLH

**DB2 LATCH - AV.EVENT/EVENTS**

The number of wait trace events processed for waits for latch contention.

**Note:** The internally defined field adjusts the original DB2 value. DB2 counts each event twice, one for the entry and one for the exit.

**Field Name:** ADLASUSC

**SYNCHRON. I/O - AVERAGE TIME/ELAPSED TIME**

The I/O elapsed time accumulated due to synchronous I/O suspensions. DB2 calculates this value by subtracting the store clock time when an agent begins waiting for a synchronous I/O from the time the agent is resumed.

**Field Name:** ADIOSUST

**SYNCHRON. I/O - AV.EVENT/EVENTS**

The total number of synchronous I/O suspensions.

**Field Name:** ADIOSUSC
Times - Class 3 - Suspensions

**DATABASE I/O - AVERAGE TIME/ELAPSED TIME**

The accumulated I/O elapsed wait time for database I/O done under this thread. This field is for synchronous I/O only. It includes synchronous read and write I/O. This value is an average.

*Field Name:* QWACAWTI

**DATABASE I/O - AV.EVENT/EVENTS**

The number of wait trace events processed for waits for database I/O under this thread.

*Field Name:* ADIOARNE

**LOG WRITE I/O - AVERAGE TIME/ELAPSED TIME**

The accumulated wait time for log write I/O.

This value is an average.

*Field Name:* QWACAWLG

**LOG WRITE I/O - AV.EVENT/EVENTS**

The number of wait trace events processed for waits for log write I/O. This value is an average.

*Field Name:* ADLWSUSC

**OTHER READ I/O - AVERAGE TIME/ELAPSED TIME**

The accumulated waiting time due to a read I/O that performed under a thread other than the one being reported. The time does not represent the total duration of the subject read I/O. It includes:

- Sequential prefetch
- List prefetch
- Sequential detection
- Synchronous read I/O performed by a thread other than the one being reported

*Field Name:* QWACAWTR

This is an *exception* field.

**OTHER READ I/O - AV.EVENT/EVENTS**

The total number of suspensions due to a read I/O performed under a thread other than the one being reported.

*Field Name:* ADARSUSC

This is an *exception* field.

**OTHER WRITE I/O - AVERAGE TIME/ELAPSED TIME**

The accumulated waiting time due to a write I/O that performed under a thread other than the one being reported. This time does not represent the total duration of the subject write I/O. It includes:

- An asynchronous write I/O
- A synchronous write I/O performed by a thread other than the one being reported

*Field Name:* QWACAWTW

This is an *exception* field.
OTHER WRTE I/O - AV.EVENT/EVENTS

The total number of suspensions due to a write I/O performed under a thread other than the one being reported. It includes:

- An asynchronous write I/O
- A synchronous write I/O performed by a thread other than the one being reported.

Field Name: ADAWSUSC

This is an exception field.

SER.TASK SWTCH - AVERAGE TIME/ELAPSED TIME

The accumulated waiting time due to a synchronous execution unit switching to DB2 services from the thread being reported. It includes:

- Open/close data set
- SYSLGRNG or SYSLGRNX update
- Commit phase 2 for read-only threads originating from TSO or batch
- Dataspace manager services
- Define data set
- Extend data set
- Delete data set
- Log I/Os for commit and abort processing

Special Considerations:
1. A probable cause for high values in this field is data set preformatting.
2. There are no service waits associated with commit phase 2 under read-only threads originating from CICS or IMS. There is a service wait for any thread doing commit phase 2 after an update.
3. There is no overlap between the elapsed time reported in this field and the other class 3 elapsed times.

Field Name: ADSTSUST

This is an exception field.

SER.TASK SWTCH - AV.EVENT/EVENTS

The total number of suspensions due to a synchronous execution unit switching to DB2 services from the thread being reported.

Field Name: ADSTSUSC

This is an exception field.

UPDATE COMMIT - AVERAGE TIME/ELAPSED TIME

The accumulated wait time because of synchronous execution unit switch for DB2 Phase 2 commit, abort, or deallocation. This includes wait time for Phase 2 commit Log writes and database writes for LOB with LOG NO. For data sharing environment Page P-locks unlocks for updated pages and GBP writes.

Field Name: QWACAWTE

UPDATE COMMIT - AV.EVENT/EVENTS

The number of wait trace events processed for waits for synchronous execution unit switching for commit or abort.

This value is an average.
**Field Name:** ADSTARNS  
**OPEN/CLOSE - AVERAGE TIME/ELAPSED TIME**  
Accumulated waiting time for a synchronous execution unit switch to the DB2 OPEN/CLOSE data set service for the HSM recall service.  
This value is an average.

**Field Name:** QWAXOCSE  
**OPEN/CLOSE - AV.EVENT/EVENTS**  
The number of wait trace events processed for waits for synchronous execution unit switching to the open/close service.  
This value is an average.

**Field Name:** ADOCSUSC  
**SYSLGRNG REC - AVERAGE TIME/ELAPSED TIME**  
Accumulated wait time for a synchronous execution unit switch to the DB2 SYSLGRNG recording service. This service is sometimes used for Level ID checking for downlevel detection.  
This value is an average.

**Field Name:** QWAXSLSE  
**SYSLGRNG REC - AV.EVENT/EVENTS**  
The number of wait trace events processed for waits for synchronous execution unit switching to the SYSLGRNG recording service.  
This value is an average.

**Field Name:** ADSLSUSC  
**EXT/DEL/DEF - AVERAGE TIME/ELAPSED TIME**  
Accumulated wait time for a synchronous execution unit switch to the DB2 data space manager services. This includes DEFINE DATA SET, EXTEND DATA SET, DELETE DATA SET, RESET DATA SET, and VSAM CATALOG ACCESS.  
This value is an average.

**Field Name:** QWAXDSSE  
**EXT/DEL/DEF - AV.EVENT/EVENTS**  
The number of wait trace events processed for waits for synchronous execution unit switching to the data space manager service tasks.  
This value is an average.

**Field Name:** ADDSSUSC  
**OTHER SERVICE - AVERAGE TIME/ELAPSED TIME**  
The VSAM catalog update. In the distributed environment, it includes the waiting time for the response from the server system.

**Field Name:** QWAXOTSE  
**OTHER SERVICE - AV.EVENT/EVENTS**  
The number of wait trace events processed for waits for synchronous execution unit switching to other service tasks.
This value is an average.

**Field Name:** ADOTSUSC

**ARC.LOG(QUIES) - AVERAGE TIME/ELAPSED TIME**

The accumulated waiting time due to the processing of ARCHIVE LOG MODE(QUIESCE) commands.

This time does not represent the time required to perform the entire command.

**Field Name:** QWAXALOG

This is an exception field.

**ARC.LOG(QUIES) - AV.EVENT/EVENTS**

The total number of suspensions due to the processing of ARCHIVE LOG MODE(QUIESCE) commands.

This counter belongs to class 1 (not to class 3 like the rest of the fields in this section) but it is shown here to be adjacent to the archive log quiesce suspension time, which is in class 3.

**Field Name:** ADALSUSC

This is an exception field.

**LOG READ - AV.EVENT/EVENTS**

The number of wait trace events processed for archive reads, active reads, and active log prefetch reads.

**Field Name:** ADLRSUSC

This is an exception field.

**DRAIN LOCK - AVERAGE TIME/ELAPSED TIME**

The accumulated waiting time for a drain lock. This is the time the requester is suspended while waiting to acquire the drain lock.

**Field Name:** QWAXAWDR

This is an exception field.

**DRAIN LOCK - AV.EVENT/EVENTS**

The total number of suspensions due to drain lock processing.

**Field Name:** ADDRSUSC

This is an exception field.

**CLAIM RELEASE - AVERAGE TIME/ELAPSED TIME**

The accumulated waiting time for a drain waiting for claims to be released. After the drain lock is acquired, the drainer must wait for claim holders to release the object.

**Field Name:** QWAXAWCL

This is an exception field.

**CLAIM RELEASE - AV.EVENT/EVENTS**

The total number of suspensions until the claims are released.

**Field Name:** ADCMSUSC
This is an exception field.

PAGE LATCH - AVERAGE TIME/ELAPSED TIME

In the data sharing environment, within the same member, the first thread gets a P-lock (such as: Index leaf page P-Lock or P-Lock for Space map page or data page P-lock for Row level locking). With a high number of concurrent threads, for subsequent threads in the same member for the same resource, contention is reported as encountering a page latch contention. Randomizing the Index key helps minimizing page latch contentions for the Index leaf page. The Member Cluster option reduces page latch contention for a Space map page.

Field Name: QWACAWTP

PAGE LATCH - AV.EVENT/EVENTS

The total number of suspensions due to page latch contentions.

Field Name: ADPGSUSC

NOTIFY MSGS - AVERAGE TIME/ELAPSED TIME

The accumulated elapsed waiting time due to suspensions caused by sending notify messages to other members in the data sharing group. Messages are sent, for example, when the database descriptors are changed due to DDL.

Field Name: QWACAWTG

NOTIFY MSGS - AV.EVENT/EVENTS

The number of suspensions caused by sending messages to other members in the data sharing group. This value is only calculated if accounting class 3 is active and DB2 is a member of a data sharing group.

Field Name: ADNOSUSC

GLOBAL CONTENTION - AVERAGE TIME/ELAPSED TIME

The total accumulated waiting time caused by the suspension of IRLM lock requests due to global lock contention in a data sharing environment that require intersystem communication to resolve.

Field Name: ADGCSUST

GLOBAL CONTENTION - AV.EVENT/EVENTS

The number of suspensions caused by global lock contention. This value is only calculated if accounting class 3 is active and DB2 is a member of a data sharing group.

Field Name: ADGCSUSC

COMMIT PH1 WRITE I/O - AVERAGE TIME/ELAPSED TIME

The accumulated time waiting for phase 1 commit write I/O. An example for this suspension is LOB Table Space with LOG NO Phase 1 commit database synchronous write I/O processing.

Field Name: QWAXAWFC

COMMIT PH1 WRITE I/O - AV.EVENT/EVENTS

The total number of wait trace events for commit phase 1 I/O.

Field Name: ADFCSUSC
Times - Class 3 - Suspensions

ASYNCH CF REQUESTS - AVERAGE TIME/ELAPSED TIME
The accumulated wait time for IXLCACHE and IXLFCOMP requests.
Field Name: QWAXIXLT

ASYNCH CF REQUESTS - AV.EVENT/EVENTS
The number of IXLCACHE and IXLFCOMP asynchronous requests.
Field Name: ADIXSUSC

TCP/IP LOB XML - AVERAGE TIME/ELAPSED TIME
The accumulated wait time for TCP/IP LOB and XML (storing large object and XML) materialization.
Field Name: QWACALBW

TCP/IP LOB XML - AV EVENT/EVENTS
The number of wait trace events processed for waits for TCP/IP LOB and XML materialization.
Field Name: ADLMSUSC

ACCELERATOR - AVERAGE TIME/ELAPSED TIME
The accumulated wait time for requests to an accelerator.
Field Name: QWACAACW

ACCELERATOR - AV EVENT/EVENTS
The total number of suspensions due to a request to an accelerator.
Field Name: ADAASUSC

AUTONOMOUS PROCEDURE - AVERAGE TIME/ELAPSED TIME
The accumulated time waiting for autonomous transactions to complete.
Field Name: AATXSUST

AUTONOMOUS PROCEDURE - AV EVENT/EVENTS
The value depends on the record type:
1. For non-rollup records, this value is the number of autonomous transactions executed.
2. For parallel query rollup records, this value is 0.
3. For a DDF/RRSAF rollup records, this value is the number of autonomous transactions executed. These transactions are NOT counted in QWACPCNT.
4. For autonomous transaction rollup records, this value is 0.
Field Name: AATCOUNT

PQ SYNCHRONIZATION - AVERAGE TIME/ELAPSED TIME
The accumulated time waiting for parallel queries to synchronize between parent and child tasks.
Field Name: AWPQSST

PQ SYNCHRONIZATION - AV EVENT/EVENTS
The number of times the parallel query processing had to suspend because it was waiting for the synchronization of parent or child.
Field Name: AWPQSSC
Times - Class 3 - Suspensions

TOTAL CLASS 3 - AVERAGE TIME/ELAPSED TIME
The waiting time for all types of class 3 suspensions by the originating task and parallel tasks, if parallelism is employed.

**Field Name:** ADTSUST
This is an *exception* field.

TOTAL CLASS 3 - AV.EVENT/EVENTS
The total number of class 3 suspensions.

**Field Name:** ADTSUSC
This is an *exception* field.
Times - Class 5 - IFI Time

This topic shows detailed information about “Accounting - Times - Class 5 - IFI Time”.

This block shows information for the Instrumentation Facility Interface (IFI) class 5.

The following example shows both layouts, the report on the left, and the trace layout on the right.

Accounting - Times - Class 5 - IFI Time

The field labels shown in the following sample layout of “Accounting - Times - Class 5 - IFI Time” are described in the following section.

ELAPSED TIME

The accumulated elapsed time for processing IFI calls. This field is only calculated if accounting class 5 is active.

Field Name: QIFAAIET

CP CPU TIME

The accumulated CPU time processing IFI calls.

This field is only calculated when accounting class 5 is active.

Field Name: AIFAAITT

NONNESTED

The accumulated CPU time spent processing IFI calls. This is the same as the TCB time (class 5).

This field is only calculated if accounting class 5 is active.

Field Name: QIFAAITT

This is an exception field.

DCAPT.DESCRIPT.
Times - Class 5 - IFI Time

The accumulated elapsed time for processing data capture describes. Data capture describes occur only during IFI read requests for IFCID 185. This time is a subset of the log extraction time.

**Field Name:** QIFAAMBT

This is an *exception* field.

**LOG EXTRACT.**

The accumulated elapsed time for extracting log records for tables defined with DATA CAPTURE CHANGES. This time is a subset of the class 5 elapsed time.

**Field Name:** QIFAAMLT

This is an *exception* field.
Times - Class 7 - CP CPU Distribution

This topic shows detailed information about “Accounting - Times - Class 7 - CP CPU Distribution”.

This block shows the distribution of the class 7 CP CPU time among all packages.

The following example shows both layouts, first the report layout, followed by the trace layout.

**Accounting - Times - Class 7 - CP CPU Distribution**

The field labels shown in the following sample layout of “Accounting - Times - Class 7 - CP CPU Distribution” are described in the following section.

Report:

<table>
<thead>
<tr>
<th>PROGRAM NAME</th>
<th>CLASS 7 CP CPU TIME CONSUMERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCRCZ043</td>
<td></td>
</tr>
<tr>
<td>CCRCZ063</td>
<td></td>
</tr>
<tr>
<td>CECEZ011</td>
<td></td>
</tr>
<tr>
<td>CECEZ012</td>
<td></td>
</tr>
<tr>
<td>CECEZ072</td>
<td></td>
</tr>
<tr>
<td>CECEZ074</td>
<td></td>
</tr>
<tr>
<td>CFS041N</td>
<td>=&gt; 96%</td>
</tr>
<tr>
<td>CPDCZG17</td>
<td></td>
</tr>
<tr>
<td>CPDCZG18</td>
<td></td>
</tr>
<tr>
<td>CPDCZG19</td>
<td></td>
</tr>
<tr>
<td>CPDCZ0A5</td>
<td></td>
</tr>
<tr>
<td>CPDCZ0BE</td>
<td>=&gt; 4%</td>
</tr>
</tbody>
</table>

Trace:

<table>
<thead>
<tr>
<th>PROGRAM NAME</th>
<th>CLASS 7 CP CPU TIME CONSUMERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSNTEP2</td>
<td>=&gt; 3%</td>
</tr>
<tr>
<td><em>ROLLUP</em></td>
<td>=&gt; 97%</td>
</tr>
</tbody>
</table>

**PROGRAM NAME**

The program name (package ID or DBRM name).

In the case of rollup data (Accounting data of DDF/RRSAF threads and parallel tasks accumulated by DB2), the following value is shown:

- *ROLSUM* for DB2 10 or later
- *ROLLUP* for DB2 versions prior to DB2 10

**Field Name:** QPACPKID

This is an exception field.

**CLASS 7 CP CPU TIME CONSUMERS**

The ratio of the class 7 CP CPU time, expressed as a percentage of the total class 7 CP CPU time of all programs.

**Field Name:** ARATCL7C
Times - Class 7 - Elapsed Time Distribution

This topic shows detailed information about “Accounting - Times - Class 7 - Elapsed Time Distribution”.

This block shows the distribution of the class 7 elapsed time among all programs.

**Note:** The name of this block depends on the DB2 version that is installed:
- **DB2 10 or later:** Times - Class 7 - Elapsed Time Distribution
- **Prior to DB2 10:** Times - Class 7 - Distribution

The following example shows both layouts, first the report layout, followed by the trace layout.

**Accounting - Times - Class 7 - Elapsed Time Distribution**

The field labels shown in the following sample layout of “Accounting - Times - Class 7 - Elapsed Time Distribution” are described in the following section.

**Report:**

<table>
<thead>
<tr>
<th>PROGRAM NAME</th>
<th>CLASS 7 ELAPSED TIME CONSUMERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCRZ043</td>
<td></td>
</tr>
<tr>
<td>CCRZ063</td>
<td></td>
</tr>
<tr>
<td>CECEZ011</td>
<td></td>
</tr>
<tr>
<td>CECEZ012</td>
<td></td>
</tr>
<tr>
<td>CECEZ072</td>
<td></td>
</tr>
<tr>
<td>CECEZ074</td>
<td></td>
</tr>
<tr>
<td>CFS041N</td>
<td>============&gt; 79%</td>
</tr>
<tr>
<td>CPDCZG17</td>
<td></td>
</tr>
<tr>
<td>CPDCZG18</td>
<td></td>
</tr>
<tr>
<td>CPDCZG19</td>
<td></td>
</tr>
<tr>
<td>CPDCZ0A5</td>
<td></td>
</tr>
<tr>
<td>CPDCZ0BE</td>
<td>==========&gt; 21%</td>
</tr>
</tbody>
</table>

**Trace:**

<table>
<thead>
<tr>
<th>PROGRAM NAME</th>
<th>CLASS 7 ELAPSED TIME CONSUMERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSNTEP2</td>
<td>=&gt; 3%</td>
</tr>
<tr>
<td><em>ROLLUP</em></td>
<td>============&gt; 97%</td>
</tr>
</tbody>
</table>

**PROGRAM NAME**

The program name (package ID or DBRM name).

In the case of rollup data (Accounting data of DDF/RRSAF threads and parallel tasks accumulated by DB2), the following value is shown:
- *ROLSUM* for DB2 10 or later
- *ROLLUP* for DB2 versions prior to DB2 10

**Field Name:** QPACPKID

This is an exception field.

**CLASS 7 ELAPSED TIME CONSUMERS**

The ratio of the class 7 elapsed time, expressed as a percentage of the total class 7 elapsed time of all programs.

**Note:** The name of this field depends on the DB2 version that is installed:
Times - Class 7 - Elapsed Time Distribution

- DB2 10 or later: CLASS 7 ELAPSED TIME CONSUMERS
- Prior to DB2 10: CLASS 7 CONSUMERS

Field Name: ARATCL7
Triggers

This topic shows detailed information about “Accounting - Triggers”.

For formatting reasons, OMEGAMON XE for DB2 PE shows different labels for report and trace. The following example shows both layouts, the report on the left, and the trace layout on the right.

**Accounting - Triggers**

The field labels shown in the following sample layout of “Accounting - Triggers” are described in the following section.

<table>
<thead>
<tr>
<th>Report:</th>
<th>Trace:</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRIGGERS</td>
<td>TRIGGERS</td>
</tr>
<tr>
<td>STATEMENT TRIGGER</td>
<td>STMT TRIGGER</td>
</tr>
<tr>
<td>ROW TRIGGER</td>
<td>ROW TRIGGER</td>
</tr>
<tr>
<td>SQL ERROR OCCUR</td>
<td>SQL ERROR</td>
</tr>
<tr>
<td>15.00</td>
<td>0</td>
</tr>
<tr>
<td>8.00</td>
<td>0</td>
</tr>
<tr>
<td>0.00</td>
<td>0</td>
</tr>
</tbody>
</table>

**STATEMENT TRIGGER (STMT TRIGGER)**

The number of times a statement trigger was activated.

**Field Name:** QXSTTRG

**ROW TRIGGER**

The number of times a row trigger was activated.

**Field Name:** QXROWTRG

**SQL ERROR OCCUR (SQL ERROR)**

The number of times an SQL error occurred during the execution of a triggered action. This includes errors that occur in user-defined functions or stored procedures that are called from triggers and that pass back a negative SQLCODE.

**Field Name:** QXTRGERR
Truncated Values

This topic shows detailed information about “Accounting - Truncated Values”.

OMEGAMON XE for DB2 PE can report values that are too long to fit in the space available in the report layout. When this happens, the value reported in the block is truncated. Truncated values are then listed at the end of each logical report unit, together with the full values.

The list of Truncated Values shows pairs of a truncated values (unique name) and its original full value (long name). This list can show pairs caused by long names of client end-user transaction names and workstation names. It shows the complete long name on several lines if required. A truncated value can consist of up to 32 characters, which is the length of the short name of an end-user transaction name.

The mapping between truncated and full values remains the same for multiple reports from the same input data. This mapping for multiple reports from different input data cannot be guaranteed.

The following example applies to both, the report layout and the trace layout.

Accounting - Truncated Values

The field labels shown in the following sample layout of “Accounting - Truncated Values” are described in the following section.
User-Defined Functions

User-Defined Functions

This topic shows detailed information about “Accounting - User-Defined Functions”.

This block shows information about user-defined functions.

The following example shows both layouts, the report on the left, and the trace layout on the right.

Accounting - User-Defined Functions

The field labels shown in the following sample layout of “Accounting - User-Defined Functions” are described in the following section.

<table>
<thead>
<tr>
<th>Report:</th>
<th>Trace:</th>
</tr>
</thead>
<tbody>
<tr>
<td>UDF</td>
<td>AVERAGE</td>
</tr>
<tr>
<td>--------</td>
<td>---------</td>
</tr>
<tr>
<td>EXECUTED</td>
<td>0.00</td>
</tr>
<tr>
<td>ABENDED</td>
<td>0.00</td>
</tr>
<tr>
<td>TIMED OUT</td>
<td>0.00</td>
</tr>
<tr>
<td>REJECTED</td>
<td>0.00</td>
</tr>
</tbody>
</table>

EXECUTED

The number of user-defined functions executed.

Field Name: QXCAUD

ABENDED

The number of times a user-defined function abended.

Field Name: QXCAUDAB

TIMED OUT

The number of times a user-defined function timed out while waiting to be scheduled.

Field Name: QXCAUDTO

REJECTED

The number of times a user-defined function was rejected.

Field Name: QXCAUDRJ
Chapter 6. The Accounting Save-File Utility

Use the Save-File utility to migrate and convert Accounting Save data sets into a format suitable for OMEGAMON XE for DB2 PE V5.2.

The function performed is specified in a parameter on the EXEC command.

“Migrating Data Sets” on page 6-2
This topic describes how to migrate Accounting Save data sets created by OMEGAMON XE for DB2 PE V4.2, V5.1.0, or V5.1.1 into the record format of OMEGAMON XE for DB2 PE V5.2.

“Converting Data Sets” on page 6-3
To store performance data in the Performance Database you must first convert Accounting Save data sets of OMEGAMON XE for DB2 PE V5.2 into sequential data sets that can be used by the DB2 load utility.

“Save-File Utility DD Statements” on page 6-4
This topic lists the DD statements needed for migration and conversion. All of the DD statements described here are required.
Migrating Data Sets

This topic describes how to migrate Accounting Save data sets created by OMEGAMON XE for DB2 PE V4.2, V5.1.0, or V5.1.1 into the record format of OMEGAMON XE for DB2 PE V5.2.

To migrate Accounting Save data sets:

1. Define an OMEGAMON XE for DB2 PE V5.2 VSAM data set using IDCAMS as output.
2. Use the MIGRATE function of the Save-File utility to migrate the data sets of OMEGAMON XE for DB2 PE V4.2, V5.1.0, or V5.1.1.

The RKO2SAMP library provides the sample job DGOPJAMI, which you can modify to suit your installation.

Note: Save data sets from previous versions V4.2, V5.1.0, or V5.1.1 cannot be restored or converted until migrated to OMEGAMON XE for DB2 PE V5.2 format.
Converting Data Sets

To store performance data in the Performance Database you must first convert Accounting Save data sets of OMEGAMON XE for DB2 PE V5.2 into sequential data sets that can be used by the DB2 load utility.

You can convert data sets in one of the following ways to store performance data in the Performance Database:

- You can use the SAVE subcommand with the CONVERT option to convert and save reduced data into a sequential data set that can be loaded into DB2 tables. The output of this subcommand option is a sequential data set, that is specified and requested in SYSIN. The data set attributes are:

  Organization
  - PS

  Record format
  - VB

  Record length
  - 9072

  Block size
  - 9076

  For more information refer to Reporting User’s Guide.

- You can use the converted Save-File data sets to generate CSV (comma-separated value) input-data. This CSV data can then be transferred to workstations and imported into spreadsheets to improve DB2 performance analysis using graphical representations or pivot tables. For more information refer to Reporting User’s Guide.

- You can use the CONVERT function to convert Statistics Save data sets of OMEGAMON XE for DB2 PE V5.2 into sequential data sets that can be used by the DB2 load utility. This enables you to store performance data in the Performance Database.

The RKO2SAMP library provides the sample job DGOPJACO, which converts Save data sets into sequential Save-File data sets, suitable for use with the DB2 load utility. You can modify this sample to suit your installation.

The following list shows the types of records that are created by the CONVERT function and where to find their layout descriptions in the sample library RKO2SAMP:

- General data records (DGOADSGE)
- Buffer pool data records (DGOADSBU)
- Distributed Data Facility (DDF) data records (DGOADSDF)
- Group buffer pool records (DGOADSGP)
- Package records (DGOADSPK)
- Resource Limit Facility (RLF) records (DGOADSRF)
- Accelerator records (DGOADSXC)
Save-File Utility DD Statements

This topic lists the DD statements needed for migration and conversion. All of the DD statements described here are required.

**Input**

The DDNAME of the input data set. This can be an OMEGAMON XE for DB2 PE V4.2.0, V5.1.0, or V5.1.1 Accounting Save data set for the MIGRATE function, or an OMEGAMON XE for DB2 PE V5.2 Accounting VSAM Save data set for the CONVERT function.

**Output**

The DDNAME of the output data set.

For CONVERT, allocate the data set with the following characteristics:

- **RECFM**: VB
- **LRECL**: 9072
- **BLKSIZE**: 9076

Refer to Chapter 64, “OMEGAMON XE for DB2 PE VSAM Data Sets,” on page 64-1 for details on how to specify the allocated data sets to migrate to OMEGAMON XE for DB2 PE V5.2.

**DPMLOG**

OMEGAMON XE for DB2 PE command processor messages and messages indicating exceptional processing conditions are written to DPMLOG. If DPMLOG is not specified, it is dynamically allocated to the SYSOUT message class of the job.

Allocate the data set with the following attributes:

- **RECFM**: FBA
- **LRECL**: 133
- **BLKSIZE**: 6251
Chapter 7. The Accounting File Data Set and Output Record

The FILE subcommand formats DB2 Accounting records and writes them to sequential data sets suitable for use by the DB2 load utility.

You can store unreduced Accounting data into the OMEGAMON XE for DB2 PE performance database. The performance database produces tailored reports using a reporting facility such as Query Management Facility (QMF).

You can also use the File data sets to generate CSV (comma-separated value) input-data. This CSV data can then be transferred to workstations and imported into spreadsheets to improve DB2 performance analysis using graphical representations or pivot tables. For more information refer to Reporting User’s Guide.

FILE can also be used to produce data sets containing only exception records. The following record format types are available:

- General Accounting data (for example, SQL counters, times, and locking activity)
- Buffer pool data
- DDF data
- Package data
- Group buffer pool data
- Accelerator data

The output is a sequential data set containing information from the DB2 IFCID 003 and IFCID 239 records. The parallel records are contained in the originating record. The number of records in the output are as follows:

- One record for General Accounting data
- Separate records for each buffer pool used
- Separate records for each remote location participating in the distributed activity
- Separate records for each package and DBRM executed

Descriptions of the Accounting File data sets and the fields contained can be found in the RKO2SAMP library under the following names:

**DGOADFGFGE**
General Accounting records

**DGOADFBU**
Buffer pool records

**DGOADDFDF**
DDF records

**DGOADFGP**
Group buffer pool records

**DGOADFPK**
Package records

**DGOADFXC**
Accelerator records
Part 3. The Audit Report Set

These topics provide information about the Audit reports.

Note: For an introduction to the Audit report set and general audit information refer to the Reporting User’s Guide

Chapter 8, “The Audit Summary Reports,” on page 8-1
The Audit summary reports present aggregated DB2 data. Data is accumulated and grouped by the specified OMEGAMON XE for DB2 PE identifiers.

Chapter 9, “The Audit Detail Report and the Audit Trace,” on page 9-1
Both the audit detail report and the audit trace show a detailed listing of all occurrences of the different audit types. The layout of the Audit report and trace is similar. The audit trace is sorted by timestamp, the audit detail report is sorted first by identifier, then by timestamp. Any combination of event types can be specified.

Chapter 10, “The Audit File Data Set and Output Record,” on page 10-1
The FILE subcommand formats DB2 Audit records and writes them to sequential data sets that can be loaded into DB2 tables.
Chapter 8. The Audit Summary Reports

The Audit summary reports present aggregated DB2 data. Data is accumulated and grouped by the specified OMEGAMON XE for DB2 PE identifiers.

The LEVEL subcommand option creates a basic summary report, which shows totals for the different audit types. Use the LEVEL and TYPE options to produce summary reports for the audit report types.

“Summary Report - Basic and Field Descriptions” on page 8-2
To produce a basic summary report, use the AUDIT REPORT LEVEL(SUMMARY) command without any TYPE constraints.

“Authorization Change Summary Report and Fields (AUTHCHG)” on page 8-5
This report presents all authorization change events according to the combination of OMEGAMON XE for DB2 PE identifiers specified.

“Authorization Control Summary Report and Fields (AUTHCNTL)” on page 8-7
This report presents all authorization control events according to the combination of OMEGAMON XE for DB2 PE identifiers specified.

“Authorization Failure Summary Report and Fields (AUTHFAIL)” on page 8-9
This report presents all authorization failure events according to the combination of OMEGAMON XE for DB2 PE identifiers you specified.

“DML at Bind Access Summary Report and Fields (BIND)” on page 8-12
This report presents all DML at bind access events according to the combination of OMEGAMON XE for DB2 PE identifiers specified.

“DDL Access Summary Report and Fields (DDL)” on page 8-14
This report presents all DDL access events according to the combination of OMEGAMON XE for DB2 PE identifiers specified.

“DML Access Summary Report and Fields (DML)” on page 8-16
This report presents all DML access events according to the combination of OMEGAMON XE for DB2 PE identifiers specified.

“Utility Access Summary Report and Fields (UTILITY)” on page 8-18
This presents all utility access events according to the combination of OMEGAMON XE for DB2 PE identifiers specified.
Summary Report - Basic and Field Descriptions

To produce a basic summary report, use the AUDIT REPORT LEVEL(SUMMARY) command without any TYPE constraints.

You can generate a summary report as follows:

```
: AUDIT
 REPORT
 LEVEL (SUMMARY)
:
```

Note: For bind events, specify the program name for PLANNNAME in ORDER, INCLUDE and EXCLUDE. For utility events, specify the utility name of the PLANNNAME in ORDER, INCLUDE, and EXCLUDE. The header of this summary report will, however, still show PLANNNAME.

In group-scope reports, MEMBER and SUBSYSTEM are not shown.

Levels of accumulation (Audit summary report)

The Audit summary report gives the following levels of accumulation:

**GROUP TOTAL**

The group total is printed on group-scope reports when the member value changes.

**SUBTOTAL**

If you request ordering by three identifiers, a subtotal block of two lines is printed on the change of the second-level identifier when there is more than one third-level identifier reported under it.

The first line shows the string "SUBTOTAL" in the first column.

The second line shows the name of the second identifier in the first column and the calculated data in all other columns.

**TOTAL**

If you request ordering by two or three identifiers, a total block of two lines with all applicable data is printed on the change of the first-level identifier when there is more than one second-level identifier reported under it.

The first line shows the string "TOTAL" in the first column.

The second line shows the name of the first identifier in the first column, and the calculated data in all other columns.

**GRAND TOTAL**

A grand total block of two lines with all applicable data is printed for a location when there is more than one first-level identifier reported.

The first line shows the string "GRAND TOTAL" in the first column.

The second line shows the name of the member (in a member-scope report) or the group (in a group-scope report) in the first column, and the calculated data in all other columns.
## Field description

The basic Audit summary report contains the following fields:

**OMEGAMON XE for DB2 PE identifiers**

The report can be sorted by up to five combinations of any three identifiers. These are printed in the first three columns from the left and indented in the sequence specified by ORDER.

The default is PRIMAUTH-PLANNAME.

For group-scope reports, MEMBER is automatically added as the last identifier.

The values printed in the following columns represent totals for each combination of the selected OMEGAMON XE for DB2 PE identifiers.

**TOTAL**

A total number of Audit events.

**AUTH FAILURE**

The total number of authorization failures (IFCID 140 records).

**GRANT/REVOKE**

The total number of authorization GRANTs or REVOKEs (IFCID 141).

**DDL ACCESS**

The total number of DDL operations against auditable DB2 tables (IFCID 142 records).

**DML READ ACCESS**

The total number of first READ attempts within a logical unit of work against auditable DB2 tables (IFCID 144 records).
Audit Report - Summary

**DML WRITE ACCESS**
The total number of first WRITE attempts against audited DB2 tables (IFCID 143 records).

**DML AT BIND**
The total number of statements referenced during a static or dynamic bind (IFCID 145) against auditable DB2 tables.

**AUTHID CHANGE**
The total number of initial AUTHID establishments, AUTHID changes, or attempted AUTHID changes (IFCID 055, 083, 087, and 169 records).

**UTILITY ACCESS**
The total number of times a utility was used to access a DB2 object (IFCID 024 records).
Audit Report - Authorization Change Summary (AUTHCHG)

Authorization Change Summary Report and Fields (AUTHCHG)

This report presents all authorization change events according to the combination of OMEGAMON XE for DB2 PE identifiers specified.

Use the following command to generate an authorization change summary report:

```
AUDIT
REPORT
LEVEL (SUMMARY)
TYPE (AUTHCHG)
```

Layout of a Member-Scope Audit Authorization Change Summary Report

Here is an example of a member-scope Audit Authorization Change Summary report.

<table>
<thead>
<tr>
<th>LOCATION: LOCATI_2</th>
<th>OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0)</th>
<th>PAGE: 1-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>GROUP: GROUP_1</td>
<td>AUDIT REPORT - SUMMARY</td>
<td>REQUESTED FROM: NOT SPECIFIED</td>
</tr>
<tr>
<td>MEMBER: MEMBER_2</td>
<td>AUTHORIZATION CHANGE</td>
<td>ORDER: PRIMAUTH-PLANNAME-OBJECT</td>
</tr>
<tr>
<td>SUBSYSTEM: SYS2</td>
<td>SCOPE: MEMBER</td>
<td>ACTUAL FROM: 01/30/10 04:21:44.17</td>
</tr>
<tr>
<td>DB2 VERSION: V10</td>
<td></td>
<td>TO: 01/30/10 07:19:20.25</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PRIMAUTH PLANNAME</th>
<th>ORIGINAL AUTHID</th>
<th>SET CURRENT</th>
<th>END OF IDENTIFY</th>
<th>END OF SIGNON</th>
<th>DISTRIBUTED</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTH_20 PLAN_20</td>
<td>ORAUTH30</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>XXASP33</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>+SUBTOTAL*</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>PLAN_30 ORAUTH30</td>
<td>XXASP33</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>+SUBTOTAL*</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>+TOTAL*</td>
<td></td>
<td>4</td>
<td>4</td>
<td>0</td>
<td>4</td>
<td>12</td>
</tr>
<tr>
<td>AUTH_30 PLAN_20</td>
<td>ORAUTH30</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>XXASP33</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>+SUBTOTAL*</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>PLAN_30 ORAUTH30</td>
<td>XXASP33</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>+SUBTOTAL*</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>+TOTAL*</td>
<td></td>
<td>4</td>
<td>4</td>
<td>0</td>
<td>4</td>
<td>12</td>
</tr>
</tbody>
</table>

Field description

The authorization change summary report contains the following fields:

**OMEGAMON XE for DB2 PE Identifiers**

The identifiers define the order of the Audit data reported. Up to three OMEGAMON XE for DB2 PE identifiers are printed:

The default ORDER for this report is PRIMAUTH-PLANNAME-OBJECT, where OBJECT is the original authorization ID.

For group-scope reports, MEMBER is automatically added as the last.

**ORIGINAL AUTHID**

The original value of the authorization ID as passed to the IDENTIFY or SIGNON authorization exit.

When the input record is IFCID 055 or 169, the value is the ORIGINAL AUTHID from the DB2 correlation header.
Audit Report - Authorization Change Summary (AUTHCHG)

SET CURRENT SQLID
The authorization changes due to a SET CURRENT SQLID request. The total number of IFCID 055 records for this set of identifiers.

END OF IDENTIFY
The authorization changes due to an identify request. The total number of IFCID 083 records for this set of identifiers.

END OF SIGNON
The authorization changes due to a signon. The total number of IFCID 087 records for this set of identifiers.

DISTRIBUTED TRANSLATION
The authorization changes due to distributed translation. The total number of IFCID 169 records for this set of identifiers.

TOTAL
All authorization changes. The total number of IFCIDs 55, 83, 87, and 169 for this set of identifiers.
Authorization Control Summary Report and Fields (AUTHCNTL)

This report presents all authorization control events according to the combination
of OMEGAMON XE for DB2 PE identifiers specified.

Use the following command to generate an authorization control summary report.

```
AUDIT
REPORT
LEVEL (SUMMARY)
TYPE (AUTHCNTL)
```

Layout of a Member-Scope Audit Authorization Control Summary Report

The layout of this report varies slightly depending if it is a member or group-scope report.

<table>
<thead>
<tr>
<th>LOCATION: LOCATI_2</th>
<th>OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0)</th>
</tr>
</thead>
<tbody>
<tr>
<td>GROUP: GROUP_1</td>
<td>REQUESTED FROM: NOT SPECIFIED</td>
</tr>
<tr>
<td>MEMBER: MEMBER_2</td>
<td>TO: NOT SPECIFIED</td>
</tr>
<tr>
<td>SUBSYSTEM: SYS2</td>
<td>ORDER: PRIMAUTH-PLANNAME-OBJECT</td>
</tr>
<tr>
<td>DB2 VERSION: V10</td>
<td>SCOPE: MEMBER</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>OBJECT</th>
<th>PRIMAUTH PLANNAME</th>
<th>TYPE</th>
<th>GRANTS</th>
<th>REVOKES</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTH_20</td>
<td>PLAN_20</td>
<td>TSPACE</td>
<td>0</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>PLAN_30</td>
<td>TSPACE</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td><em>TOTAL</em></td>
<td></td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>AUTH_30</td>
<td>PLAN_20</td>
<td>TSPACE</td>
<td>0</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>PLAN_30</td>
<td>TSPACE</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td><em>TOTAL</em></td>
<td></td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td></td>
<td><em>GRAND TOTAL</em></td>
<td></td>
<td>0</td>
<td>8</td>
<td>8</td>
</tr>
</tbody>
</table>

For group-scope reports:
- MEMBER and SUBSYSTEM are not shown on the page header
- MEMBER is not added to the identifiers specified.
- A GROUP TOTAL is shown when a member value changes.

Layout of a Group-Scope Audit Authorization Control Summary Report

Here is a sample layout of a Group-Scope Audit Authorization Control Summary report.

<table>
<thead>
<tr>
<th>LOCATION: LOCATI_2</th>
<th>OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0)</th>
</tr>
</thead>
<tbody>
<tr>
<td>GROUP: GROUP_1</td>
<td>REQUESTED FROM: NOT SPECIFIED</td>
</tr>
<tr>
<td></td>
<td>TO: NOT SPECIFIED</td>
</tr>
<tr>
<td>DB2 VERSION: V10</td>
<td>ORDER: OBJECT</td>
</tr>
<tr>
<td></td>
<td>SCOPE: GROUP</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>OBJECT</th>
<th>MEMBER</th>
<th>GRANTS</th>
<th>REVOKES</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>TSPACE</td>
<td>MEMBER_2</td>
<td>0</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>MEMBER_3</td>
<td>0</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td><em>GRAND TOTAL</em></td>
<td></td>
<td>16</td>
<td>16</td>
</tr>
</tbody>
</table>

Field description

The authorization control summary report contains the following fields:

**OMEGAMON XE for DB2 PE Identifiers**
- The identifiers define the order of the Audit data reported. Up to three identifiers are printed.
- The defaults are:
Audit Report - Authorization Control Summary (AUTHCNTL)

- For member-scope reports, PRIMAUTH-PLANNAME-OBJECT
- For group-scope reports, OBJECT

For group-scope reports, MEMBER is automatically added as the last identifier.

**OBJECT TYPE**

The DB2 object type of the GRANT or REVOKE. Possible values are:
- TSPACE
- LOBTS
- TAB/VIEW

**GRANTS**

All grant operations.

**REVOKE**

All revoke operations.

**TOTAL**

All grant/revoke operations. The total number of IFCID 141 records for this set of identifiers.
**Authorization Failure Summary Report and Fields (AUTHFAIL)**

This report presents all authorization failure events according to the combination of OMEGAMON XE for DB2 PE identifiers you specified.

Use the following command to generate an authorization failure summary report.

```
AUDIT
REPORT
  LEVEL (SUMMARY)
  TYPE (AUTHFAIL)
```

**Layout of a Member-Scope Audit Authorization Failure Summary Report**

The layout of the default member-scope authorization failure summary report, ordered by object type, source object, source owner, target object, and target owner within plan name within primary authorization ID, is shown in the following example.

```
LOCATION: LOCATI_2
GROUP: GROUP_1
MEMBER: MEMBER_2
SUBSYSTEM: SYS2
DB2 VERSION: V10

<table>
<thead>
<tr>
<th>PRIMAUTH</th>
<th>PLANNAME</th>
<th>PRIVILEGE</th>
<th>OBJECT</th>
<th>TYPE</th>
<th>SOURCE</th>
<th>OWNER</th>
<th>NAME</th>
<th>TARGET</th>
<th>TYPE</th>
<th>OWNER</th>
<th>NAME</th>
<th>TOTAL</th>
</tr>
</thead>
</table>
| AUTH_20  | PLAN_20  | SELECT    | TABLE  | SYSIBM| SYSDATABASE | SYSIBM | SYSDATABASE | 2
|          | PLAN_30  | SELECT    | TABLE  | SYSIBM| SYSDATABASE | SYSIBM | SYSDATABASE | 2
|          |          | *TOTAL*   |        |      |          |       |      | 4
| AUTH_30  | PLAN_20  | INSERT    | TABLE  | SYSIBM| SYSDATABASE | SYSIBM | SYSDATABASE | 2
|          | PLAN_30  | INSERT    | TABLE  | SYSIBM| SYSDATABASE | SYSIBM | SYSDATABASE | 2
|          |          | *TOTAL*   |        |      |          |       |      | 4
|          |          | *GRAND TOTAL* |    |      |          |       |      | 8
```

**Layout of a Group-Scope Audit Authorization Failure Summary Report**

Here is a sample layout of a Group-Scope Audit Authorization Failure Summary report.

```
LOCATION: LOCATI_2
GROUP: GROUP_1
DB2 VERSION: V10

<table>
<thead>
<tr>
<th>PRIVILEGE</th>
<th>OBJECT</th>
<th>TYPE</th>
<th>SOURCE</th>
<th>OWNER</th>
<th>NAME</th>
<th>TARGET</th>
<th>TYPE</th>
<th>OWNER</th>
<th>NAME</th>
<th>MEMBER</th>
<th>TOTAL</th>
</tr>
</thead>
</table>
| SELECT    | TABLE  | SYSIBM| SYSDATABASE | SYSIBM | SYSDATABASE | MEMBER_2 | 8
|          |        |      |          |       |      |        |      |       |      |        |       |
| *GRAND TOTAL* |       |     |          |       |      |        |      |       |      |        |       |
```

**Field description**

The authorization failure summary report contains the following fields:

**OMEGAMON XE for DB2 PE identifiers**

The identifiers define the order of the Audit data reported. Up to three identifiers are printed.

The defaults are:
- For member-scope reports, PRIMAUTH-PLANNAME-OBJECT
- For group-scope reports, OBJECT
Audit Report - Authorization Failure Summary

For group-scope reports, MEMBER is automatically added as the last identifier.

PRIVILEGE
The privilege that was checked. Possible values are provided in the DB2 macro DSNDQW02.

OBJECT TYPE
The DB2 object type. Possible values are:

- BUFFER
  Buffer Pool

- COLLECT
  Collection

- DATABASE
  Database

- DISTTYPE
  Distinct Type

- FUNCTION
  Function

- PACKAGE
  Package

- SCHEMA
  Schema

- PROCEDUR
  Procedure

- APPLPLAN
  Application Plan

- LOBTS
  Large Object Table Space

- STOGROUP
  Storage Group

- TAB/VIEW
  Table or View

- USERAUTH
  System privileges, such as SYSADM or SYSOPR

- SEQUENCE
  Sequence

SOURCE OBJECT OWNER
If the object type is USERAUTH and the privilege is CREATE ALIAS, this is the qualifier of the alias being created. N/A is printed when the privilege is any other value.

If the object type is not USERAUTH, this is the qualifier of the object against which the authorization was checked.

SOURCE OBJECT NAME
If the object type is USERAUTH and the privilege is CREATEALIAS, CREATEDBA, CREATEDBC, or CREATESG this is the name of the alias, or object being created. N/A is printed when the privilege has any other value.
Audit Report - Authorization Failure Summary

If the object type is not USERAUTH, this is the name of the object against which the authorization was checked.

**TARGET OBJECT OWNER**

This is the qualifier of the object being created. It is valid when the Privilege is CREATE TABLE or CREATE INDEX. It is also valid for an authorization check against the following privileges:

- CREATE VIEW
- SELECT
- INSERT
- DELETE
- UPDATE

Otherwise N/A is printed.

**TARGET OBJECT NAME**

This is the qualifier of the object being created. It is valid when the Privilege is CREATE TABLE or CREATE INDEX. It is also valid for an authorization check against the following privileges:

- CREATE VIEW
- SELECT
- INSERT
- DELETE
- UPDATE

Otherwise N/A is printed.

**TOTAL**

All authorization failures. The total number of IFCID 140 records for this set of identifiers.
Audit Report - DML at Bind Summary

DML at Bind Access Summary Report and Fields (BIND)

This report presents all DML at bind access events according to the combination of OMEGAMON XE for DB2 PE identifiers specified.

Use the following command to generate a DML at bind access summary report.

```language
AUDIT
REPORT
LEVEL (SUMMARY)
TYPE (BIND)
```

Layout of a Member-Scope Audit DML at Bind Access Summary Report

For ORDER, INCLUDE, and EXCLUDE, the program name is used for PLANNAME.

<table>
<thead>
<tr>
<th>LOCATION: LOCATI_2</th>
<th>GROUP: GROUP_1</th>
<th>SUBSYSTEM: SYS2</th>
<th>DB2 VERSION: V10</th>
</tr>
</thead>
<tbody>
<tr>
<td>GROUP: MEMBER_2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DATABASE TABLEID</td>
<td>TOTAL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Auth_20 DSNESM68 DBASE1 7</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Auth_30 DSNESM68 DBASE1 7</td>
<td>4</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*GRAND TOTAL* 8

Layout of a Group-Scope Audit DML at Bind Access Summary Report

Here is a sample layout of a Group-Scope Audit DML at Bind Access Summary report.

<table>
<thead>
<tr>
<th>LOCATION: LOCATI_2</th>
<th>GROUP: GROUP_1</th>
<th>DB2 VERSION: V10</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATABASE TABLEID</td>
<td>MEMBER</td>
<td>1ST READ 1ST WRITE TOTAL</td>
</tr>
<tr>
<td>DBASE1 7 MEMBER_2 8 8 16</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DBASE1 7 MEMBER_3 8 8 16</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*GRAND TOTAL* 16 16 32

Field description

The DML at bind access summary report contains the following fields:

**OMEGAMON XE for DB2 PE identifiers**

The identifiers define the order of the Audit data reported. Up to three identifiers are printed.

The defaults are:

- For member-scope reports, PRIMAUTH-PLNAME-OBJECT
- For group-scope reports, OBJECT

For group-scope reports, MEMBER is considered the last ORDER identifier and is automatically added as the second, third, or fourth identifier.

**DATABASE**

Either the name of the database that contains the auditable DB2 table, or the internal DB2 identification (DBID) of the database that contains the auditable DB2 table.
**Audit Report - DML at Bind Summary**

**Note:** DATABASE is a repeating field and can have more than one entry on the report.

**TABLEID**

The object identifier (OBID) of the auditable DB2 table.

**Note:** TABLEID is a repeating field and can have more than one entry on the report.

**TOTAL**

All DML at bind events. The total number of IFCID 145 records for this set of identifiers.
Audit Report - DDL Access Summary

DDL Access Summary Report and Fields (DDL)

This report presents all DDL access events according to the combination of OMEGAMON XE for DB2 PE identifiers specified.

Use the following command to generate a DDL access summary report.

```
AUDIT REPORT
LEVEL (SUMMARY)
TYPE (DDL)
```

Layout of a Member-Scope Audit DDL Access Summary Report

```
LOCATION: LOCATI_2
GROUP: GROUP_1
SUBSYSTEM: SYS2
DB2 VERSION: V10

---------- OBJECT ----------
PRIMAUTH PLANNAME OWNER NAME CREATE DROP ALTER TOTAL
-------- ------------------ -------- -------- -------- -------- --------
JUB DSNREXX COLUMN MASK 1 0 0 1
ROW PERMISSION 1 0 0 1
JUB AUDTB1 0 0 1 1
JUB AUDTB2 0 0 1 1
+SUBTOTAL+ 2 0 2 4
DSNREXY COLUMN MASK 1 0 0 1
ROW PERMISSION 1 0 0 1
JUB AUDTB1 0 0 1 1
JUB AUDTB2 0 0 1 1
+SUBTOTAL+ 2 0 2 4
+TOTAL+ 4 0 4 8
KUC DSNREXX COLUMN MASK 1 0 0 1
ROW PERMISSION 1 0 0 1
KUC AUDTB1 0 0 1 1
KUC AUDTB2 0 0 1 1
+SUBTOTAL+ 2 0 2 4
DSNREXY COLUMN MASK 1 0 0 1
ROW PERMISSION 1 0 0 1
KUC AUDTB1 0 0 1 1
KUC AUDTB2 0 0 1 1
+SUBTOTAL+ 2 0 2 4
+TOTAL+ 4 0 4 8

*GRAND TOTAL* 0 28 0 28
```

Layout of a Group-Scope Audit DDL Access Summary Report

Here is a sample layout of a Group-Scope Audit DDL Access Summary report.

```
LOCATION: LOCATI_2
GROUP: GROUP_1
DB2 VERSION: V10

---------- OBJECT ----------
OWNER NAME MEMBER CREATE DROP ALTER TOTAL
-------- -------------------- -------- -------- -------- --------
COLUMN MASK MEMBER_2 1 0 0 1
MEMBER_3 1 0 0 1
+TOTAL+ 2 0 0 2
COLUMN MASK MEMBER_2 1 0 0 1
MEMBER_3 1 0 0 1
+TOTAL+ 2 0 0 2
XXASP09 NDEPT MEMBER_2 0 7 0 7
MEMBER_3 0 7 0 7
+TOTAL+ 0 14 0 14
XXASP09 NHEMP MEMBER_2 0 7 0 7
MEMBER_3 0 7 0 7
+TOTAL+ 0 14 0 14
+GRAND TOTAL+ 0 28 0 28
```
**Field description**

The DDL access summary report contains the following fields:

**OMEGAMON XE for DB2 PE identifiers**

The identifiers define the order of the Audit data reported. Up to three identifiers are printed.

The defaults are:

- For member-scope reports, PRIMAUTH-PLANNAME-OBJECT
- For group-scope reports, OBJECT

For group-scope reports, MEMBER is considered the last ORDER identifier and is automatically added as the second, third, or fourth identifier.

**OBJECT OWNER**

The user identification of the owner of the audited object table accessed. It is blank for IFCID 271.

**OBJECT NAME**

The name of the accessed audited DB2 table, the ROW PERMISSION, or the COLUMN MASK objects that have been created, dropped, or altered.

**CREATE**

All create object operations.

**ALTER**

All create object operations.

**DROP**

All drop object operations.

**TOTAL**

All DDL access operations. The total number of IFCID 142 or IFCID 271 records for this set of identifiers.
Audit Report - DML Access Summary (DML)

DML Access Summary Report and Fields (DML)

This report presents all DML access events according to the combination of OMEGAMON XE for DB2 PE identifiers specified.

Use the following command to generate a DML access summary report.

```
AUDIT REPORT
LEVEL (SUMMARY)
TYPE (DML)
```

Layout of a Member-Scope Audit DML Access Summary Report

Here is a sample layout of a Member-Scope Audit DML Access Summary report.

```
LOCATION: LOCATI_2 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-6
GROUP: GROUP_1 AUDIT REPORT - SUMMARY REQUESTED FROM: NOT SPECIFIED
MEMBER: MEMBER_2 DML ACCESS TO: NOT SPECIFIED
SUBSYSTEM: SYS2 ORDER: PRIMAUTH-PLANNAME-OBJECT ACTUAL FROM: 01/30/10 04:21:44.17
DB2 VERSION: V10 SCOPE: MEMBER

PRIMAUTH PLANNAME DATABASE PAGESET TABLEID 1ST READ 1ST WRITE TOTAL
-------- -------- -------- -------- -------- --------- --------
AUTH_20 PLAN_20 DBASE1 PSET1 7 2 2 4
   PLAN_30 DBASE1 PSET1 7 2 2 4
*TOTAL*                                        4 4 8

AUTH_30 PLAN_20 DBASE1 PSET1 7 2 2 4
   PLAN_30 DBASE1 PSET1 7 2 2 4
*TOTAL*                                        4 4 8

*GRAND TOTAL*                                 8 8 16
```

Field description

The DML access summary report contains the following fields:

OMEGAMON XE for DB2 PE Identifiers
The identifiers define the order of the Audit data reported. Up to three identifiers are printed.

The defaults are:
- For member-scope reports, PRIMAUTH-PLANNAME-OBJECT
- For group-scope reports, OBJECT

For group-scope reports, MEMBER is considered the last ORDER identifier and is automatically added as the second, third, or fourth identifier.

DATABASE
The name of the database that contains the auditable DB2 table. If the database name is unavailable, the decimal DBID is printed.
Audit Report - DML Access Summary (DML)

**PAGESET**
The name of the page set that contains the auditable DB2 table. If the page set name is unavailable, the decimal PSID is printed. If neither of these values is present, N/P is printed.

**TABLEID**
The object identifier (OBID) of the table, if applicable, associated with the access.

**1ST READ**
The total number of first read attempts within a logical unit of work against auditable DB2 tables.

**1ST WRITE**
The total number of first write attempts against audited DB2 tables.

**TOTAL**
All DML access operations. The total number of IFCID 143 and 144 records for this set of identifiers.
Audit Report - Utility Access Summary (UTILITY)

Utility Access Summary Report and Fields (UTILITY)

This presents all utility access events according to the combination of OMEGAMON XE for DB2 PE identifiers specified.

Use the following command to generate a utility access summary report.

```
AUDIT
REPORT
   LEVEL (SUMMARY)
   TYPE (UTILITY)
```

Note: For ORDER, INCLUDE and EXCLUDE, the utility name is used for PLANNAME.

Layout of a Member-Scope Audit Utility Access Summary Report

Here is a sample layout of a Member-Scope Audit Utility Access Summary report.

```
LOCATION: LOCATI_2
GROUP: GROUP_1
MEMBER: MEMBER_2
SUBSYSTEM: SYS2
DB2 VERSION: V10
PRIMAUTH UTILNAME DATABASE PAGESET TOTAL
-------- -------- -------- -------- --------
AUTH_20 LOAD DBASE1 PSET1 4
AUTH_30 LOAD DBASE1 PSET1 4
+GRAND TOTAL+ 8
```

Layout of a Group-Scope Audit Utility Access Summary Report

The layout of the default group-scope utility access summary report, ordered by member within database and page set, is shown in “Layout of a Group-Scope Audit Utility Access Summary Report.”

```
LOCATION: LOCATI_2
GROUP: GROUP_1
DB2 VERSION: V10
DATE: 07/19/20.25
PRIMAUTH UTILNAME DATABASE PAGESET MEMBER TOTAL
-------- -------- -------- -------- --------
DATABASE PAGESET MEMBER TOTAL
-------- -------- -------- -------- --------
DBASE1 PSET1 MEMBER_2 8
MEMBER_3 8
MEMBER_3 8
+GRAND TOTAL+ 16
```

Field description

The utility access summary report contains the following fields:

OMEGAMON XE for DB2 PE Identifiers

The identifiers define the order of the Audit data reported. Up to three identifiers are printed.

The defaults are:

- For member-scope reports, PRIMAUTH-PLANNAME-OBJECT
- For group-scope reports, OBJECT

For group-scope reports, MEMBER is considered the last ORDER identifier and is automatically added as the second, third, or fourth identifier.
Audit Report - Utility Access Summary (UTILITY)

DATABASE
The name of the database that contains the auditable DB2 object or the
decimal DBID of that database.

PAGESET
The name or the decimal PSID of the page set that contains the DB2 object.

TOTAL
All utility access operations. Total number of IFCID 024 records for this set
of identifiers.
Accounting Report
Chapter 9. The Audit Detail Report and the Audit Trace

Both the audit detail report and the audit trace show a detailed listing of all occurrences of the different audit types. The layout of the Audit report and trace is similar. The audit trace is sorted by timestamp, the audit detail report is sorted first by identifier, then by timestamp. Any combination of event types can be specified.

Use the following command to generate an audit detail report.

```
AUDIT REPORT
  LEVEL (DETAIL)
```

Use the following command to generate an audit trace.

```
AUDIT TRACE
```

"Example of a Member-Scope Audit Detail Report and Trace (Type AUTHCNTL)” on page 9-2
This section shows examples of a Member-Scope Audit Detail Report and Trace (Type AUTHCNTL).

"Example of a Member-Scope Audit Detail Report and Trace (Type AUTHFAIL)” on page 9-6
This section shows examples of a Member-Scope Audit Detail Report and Trace (Type AUTHFAIL).

"Field Descriptions (Audit Detail Report and Audit Trace)” on page 9-11
"Authorization Change Detail (Type AUTHCHG)” on page 9-12
The following sections list the various types of authorization changes that can be printed. They describe the fields if you select AUTHCHG. The types are sorted by IFCID.

"Authorization Control Detail (Type AUTHCNTL)” on page 9-22
This topic shows detailed information about "Authorization Control Detail (Type AUTHCNTL)".

"Authorization Failure Detail (Type AUTHFAIL)” on page 9-26
This topic shows detailed information about "Authorization Failure Detail (Type AUTHFAIL)".

"DML at Bind Access Detail (Type BIND)” on page 9-28
This topic shows detailed information about "DML at Bind Access Detail (Type BIND)".

"DDL Access Detail (Type DDL)” on page 9-30
This topic shows the various types of DDL Access detail that can be printed. It describes the corresponding fields if you select DDL. The types are sorted by IFCID.

"DML Access Detail (Type DML)” on page 9-34
This topic shows a sample and the field description of “DML Access Detail (Type DML)".

"Utility Access Detail (Type UTILITY)” on page 9-35
This topic shows a sample and the field description of the “Utility Access Detail (Type UTILITY)".

© Copyright IBM Corp. 1985, 2013
Example of a Member-Scope Audit Detail Report and Trace (Type AUTHCNTRL)

This section shows examples of a Member-Scope Audit Detail Report and Trace (Type AUTHCNTRL).

<table>
<thead>
<tr>
<th>PLANNAME</th>
<th>CONNECT</th>
<th>TIMESTAMP</th>
<th>TYPE</th>
<th>DETAIL</th>
</tr>
</thead>
<tbody>
<tr>
<td>JUB</td>
<td></td>
<td>13:04:11.38</td>
<td>AUTHCNTRL</td>
<td>GRANTOR: JUB OWNER TYPE: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TEXT: GRANT INSERT ON TABLE JUB.AUDTB1 TO PUBLIC</td>
</tr>
<tr>
<td>JUB</td>
<td></td>
<td>13:04:11.40</td>
<td>AUTHCNTRL</td>
<td>GRANTOR: JUB OWNER TYPE: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TEXT: GRANT UPDATE ON TABLE JUB.AUDTB1 TO PUBLIC</td>
</tr>
<tr>
<td>JUB</td>
<td></td>
<td>13:04:12.40</td>
<td>AUTHCNTRL</td>
<td>GRANTOR: JUB OWNER TYPE: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TEXT: GRANT UPDATE ON TABLE JUB.AUDTB1 TO PUBLIC</td>
</tr>
<tr>
<td>JUB</td>
<td></td>
<td>13:04:13.40</td>
<td>AUTHCNTRL</td>
<td>GRANTOR: JUB OWNER TYPE: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TEXT: GRANT UPDATE ON TABLE JUB.AUDTB1 TO PUBLIC</td>
</tr>
<tr>
<td>JUB</td>
<td></td>
<td>13:04:14.40</td>
<td>AUTHCNTRL</td>
<td>GRANTOR: JUB OWNER TYPE: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TEXT: GRANT UPDATE ON TABLE JUB.AUDTB1 TO PUBLIC</td>
</tr>
<tr>
<td>JUB</td>
<td></td>
<td>13:04:15.40</td>
<td>AUTHCNTRL</td>
<td>GRANTOR: JUB OWNER TYPE: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TEXT: GRANT UPDATE ON TABLE JUB.AUDTB1 TO PUBLIC</td>
</tr>
<tr>
<td>JUB</td>
<td></td>
<td>13:04:16.40</td>
<td>AUTHCNTRL</td>
<td>GRANTOR: JUB OWNER TYPE: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TEXT: GRANT UPDATE ON TABLE JUB.AUDTB1 TO PUBLIC</td>
</tr>
<tr>
<td>JUB</td>
<td></td>
<td>13:04:17.40</td>
<td>AUTHCNTRL</td>
<td>GRANTOR: JUB OWNER TYPE: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TEXT: GRANT UPDATE ON TABLE JUB.AUDTB1 TO PUBLIC</td>
</tr>
<tr>
<td>JUB</td>
<td></td>
<td>13:04:18.40</td>
<td>AUTHCNTRL</td>
<td>GRANTOR: JUB OWNER TYPE: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TEXT: GRANT UPDATE ON TABLE JUB.AUDTB1 TO PUBLIC</td>
</tr>
<tr>
<td>JUB</td>
<td></td>
<td>13:04:19.40</td>
<td>AUTHCNTRL</td>
<td>GRANTOR: JUB OWNER TYPE: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TEXT: GRANT UPDATE ON TABLE JUB.AUDTB1 TO PUBLIC</td>
</tr>
</tbody>
</table>

LOCATION: PM02D821GANZLANG OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-1
GROUP: N/P MEMBER: N/P SUBSYSTEM: DB21 ORDER: PRIMAUTH-PLANNAME SCOPE: MEMBER TO: 12/12/03 13:04:11.38
DB2 VERSION: V9 PRIMAUTH CORRNAME: CONNTYPE ORIGAUTH CORRMMBR: INSTANCE

LOCATION: PM02D821GANZLANG OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-2
GROUP: N/P MEMBER: N/P SUBSYSTEM: DB21 ORDER: PRIMAUTH-PLANNAME SCOPE: MEMBER TO: 12/12/03 13:04:20.40

LOCATION: STLEC1 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 2-1
GROUP: N/P MEMBER: N/P
<table>
<thead>
<tr>
<th>PLANNAME</th>
<th>TIMESTAMP</th>
<th>TYPE</th>
<th>DETAIL</th>
</tr>
</thead>
<tbody>
<tr>
<td>TSO</td>
<td>09:15:25.01</td>
<td>AUTHCNTL</td>
<td>EXECUTE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>OBJECT TYPE: PACKAGE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>AUTHID: SYSADM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SOURCE OBJECT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>QUALIFIER: C59656402C56</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NAME: DSNTEP3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TARGET OBJECT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>QUALIFIER: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NAME: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TEXT: N/P</td>
</tr>
<tr>
<td>TSO</td>
<td>09:15:25.01</td>
<td>AUTHCNTL</td>
<td>EXECUTE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>OBJECT TYPE: USERAUTH</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>AUTHID: SECADM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SOURCE OBJECT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>QUALIFIER: C59656402C56</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NAME: DSNTEP3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TARGET OBJECT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>QUALIFIER: ADMF001</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NAME: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TEXT: GRANT DBADM TO OMVSADM</td>
</tr>
<tr>
<td>TSO</td>
<td>09:15:25.01</td>
<td>AUTHCNTL</td>
<td>EXECUTE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>OBJECT TYPE: TAB/VIEW</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>AUTHID: JUB</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SOURCE OBJECT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>QUALIFIER: BA7587036AF1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NAME: JUB.AUDTB1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TARGET OBJECT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>QUALIFIER: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NAME: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TEXT: GRANT INSERT ON TABLE JUB.AUDTB1 TO PUBLIC</td>
</tr>
<tr>
<td>TSO</td>
<td>09:15:25.01</td>
<td>AUTHCNTL</td>
<td>EXECUTE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>OBJECT TYPE: TAB/VIEW</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>AUTHID: JUB</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SOURCE OBJECT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>QUALIFIER: BA7587036AF1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NAME: JUB.AUDTB1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TARGET OBJECT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>QUALIFIER: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NAME: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TEXT: GRANT UPDATE ON TABLE JUB.AUDTB1 TO PUBLIC</td>
</tr>
<tr>
<td>TSO</td>
<td>09:15:25.01</td>
<td>AUTHCNTL</td>
<td>EXECUTE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>OBJECT TYPE: TAB/VIEW</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>AUTHID: JUB</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SOURCE OBJECT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>QUALIFIER: BA7587036AF1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NAME: JUB.AUDTB1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TARGET OBJECT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>QUALIFIER: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NAME: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TEXT: GRANT UPDATE ON TABLE JUB.AUDTB1 TO PUBLIC</td>
</tr>
<tr>
<td>TSO</td>
<td>09:15:25.01</td>
<td>AUTHCNTL</td>
<td>EXECUTE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>OBJECT TYPE: TAB/VIEW</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>AUTHID: JUB</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SOURCE OBJECT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>QUALIFIER: BA7587036AF1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NAME: JUB.AUDTB1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TARGET OBJECT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>QUALIFIER: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NAME: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TEXT: GRANT UPDATE ON TABLE JUB.AUDTB1 TO PUBLIC</td>
</tr>
<tr>
<td>TSO</td>
<td>09:15:25.01</td>
<td>AUTHCNTL</td>
<td>EXECUTE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>OBJECT TYPE: TAB/VIEW</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>AUTHID: JUB</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SOURCE OBJECT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>QUALIFIER: BA7587036AF1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NAME: JUB.AUDTB1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TARGET OBJECT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>QUALIFIER: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NAME: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TEXT: GRANT UPDATE ON TABLE JUB.AUDTB1 TO PUBLIC</td>
</tr>
<tr>
<td>TSO</td>
<td>09:15:25.01</td>
<td>AUTHCNTL</td>
<td>EXECUTE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>OBJECT TYPE: TAB/VIEW</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>AUTHID: JUB</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SOURCE OBJECT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>QUALIFIER: BA7587036AF1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NAME: JUB.AUDTB1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TARGET OBJECT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>QUALIFIER: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NAME: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TEXT: GRANT UPDATE ON TABLE JUB.AUDTB1 TO PUBLIC</td>
</tr>
<tr>
<td>TSO</td>
<td>09:15:25.01</td>
<td>AUTHCNTL</td>
<td>EXECUTE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>OBJECT TYPE: TAB/VIEW</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>AUTHID: JUB</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>SOURCE OBJECT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>QUALIFIER: BA7587036AF1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NAME: JUB.AUDTB1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TARGET OBJECT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>QUALIFIER: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NAME: N/P</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TEXT: GRANT UPDATE ON TABLE JUB.AUDTB1 TO PUBLIC</td>
</tr>
</tbody>
</table>
JUB 'BLANK' BA7587036AF1 REASON: SYSCTRL SQLCODE: 0
DSNREXX DB2CALL OBJECT TYPE: TAB/VIEW
TEXT: GRANT UPDATE ON TABLE JUB.AUDTB1 TO PUBLIC

JUB JUB DB2CALL 13:04:16.40 AUTHCNTL GRANTOR: JUB OWNER TYPE: N/P SQLCODE: 0
JUB 'BLANK' BA7587036AF1 REASON: DBMAINT SQLCODE: 0
DSNREXX DB2CALL OBJECT TYPE: TAB/VIEW
TEXT: GRANT UPDATE ON TABLE JUB.AUDTB1 TO PUBLIC

JUB JUB DB2CALL 13:04:17.40 AUTHCNTL GRANTOR: JUB OWNER TYPE: N/P SQLCODE: 0
JUB 'BLANK' BA7587036AF1 REASON: SYSOPR SQLCODE: 0
DSNREXX DB2CALL OBJECT TYPE: TAB/VIEW
TEXT: GRANT UPDATE ON TABLE JUB.AUDTB1 TO PUBLIC

JUB JUB DB2CALL 13:04:18.40 AUTHCNTL GRANTOR: JUB OWNER TYPE: N/P SQLCODE: 0
JUB 'BLANK' BA7587036AF1 REASON: PACKADM SQLCODE: 0
DSNREXX DB2CALL OBJECT TYPE: TAB/VIEW
TEXT: GRANT UPDATE ON TABLE JUB.AUDTB1 TO PUBLIC

JUB JUB DB2CALL 13:04:19.40 AUTHCNTL GRANTOR: JUB OWNER TYPE: N/P SQLCODE: 0
JUB 'BLANK' BA7587036AF1 REASON: SYSADM SQLCODE: 0
DSNREXX DB2CALL OBJECT TYPE: TAB/VIEW
TEXT: GRANT UPDATE ON TABLE JUB.AUDTB1 TO PUBLIC

LOCATION: PMO2D821GANZLANG OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-2
GROUP: N/P AUDIT TRACE REQUESTED FROM: NOT SPECIFIED
MEMBER: N/P TO: NOT SPECIFIED
SUBSYSTEM: D821 ACTUAL FROM: 12/12/03 13:04:11.38
DB2 VERSION: V9 SCOPE: MEMBER PAGE DATE: 12/12/03

PLANNAME CONNECT TIMESTAMP TYPE DETAIL
-------- -------- ----------- -------- --------------------------------------------------------------------------------
ADMF001 WRKDDL TSO 09:15:25.01 AUTHCNTL AUTH TYPE: SYSADM
DSNTEP3 BATCH PRIV CHECKED: EXECUTE
AUTHID: SYSAOM
SOURCE OBJECT
QUALIFIER: DSNTEP3 NAME: N/P
TARGET OBJECT
QUALIFIER: DSNTEP3 NAME: N/P
TEXT: N/P
OBJECT TYPE: PACKAGE

ADMF001 WRKDDL TSO 09:15:25.01 AUTHCNTL AUTH TYPE: SYSADM
DSNTEP3 BATCH PRIV CHECKED: SECADM
AUTHID: ADMF001
SOURCE OBJECT
QUALIFIER: N/P NAME: N/P
TARGET OBJECT
QUALIFIER: ADMF001 NAME: N/P
TEXT: GRANT DBADM TO OMVSADM
OBJECT TYPE: USERAUTH

ADMF001 WRKDDL TSO 09:15:25.01 AUTHCNTL AUTH TYPE: SYSADM
DSNTEP3 BATCH PRIV CHECKED: SECADM
AUTHID: ADMF001
SOURCE OBJECT
QUALIFIER: N/P NAME: N/P
TARGET OBJECT
QUALIFIER: ADMF001 NAME: N/P
TEXT: GRANT DBADM TO OMVSADM
OBJECT TYPE: USERAUTH

ADMF001 WRKDDL TSO 09:15:25.01 AUTHCNTL AUTH TYPE: SYSADM
DSNTEP3 BATCH PRIV CHECKED: SECADM
AUTHID: ADMF001
SOURCE OBJECT
QUALIFIER: N/P NAME: N/P
TARGET OBJECT
QUALIFIER: ADMF001 NAME: N/P
TEXT: GRANT DBADM TO OMVSADM
OBJECT TYPE: USERAUTH

ADMF001 WRKDDL TSO 09:15:25.01 AUTHCNTL AUTH TYPE: SYSADM
DSNTEP3 BATCH PRIV CHECKED: SECADM
AUTHID: ADMF001
SOURCE OBJECT
QUALIFIER: N/P NAME: N/P
TARGET OBJECT
QUALIFIER: ADMF001 NAME: N/P
TEXT: GRANT DBADM TO OMVSADM
OBJECT TYPE: USERAUTH

ADMF001 WRKDDL TSO 09:15:25.01 AUTHCNTL AUTH TYPE: SYSADM
DSNTEP3 BATCH PRIV CHECKED: SECADM
AUTHID: ADMF001
SOURCE OBJECT
QUALIFIER: N/P NAME: N/P
TARGET OBJECT
QUALIFIER: ADMF001 NAME: N/P
TEXT: GRANT DBADM TO OMVSADM
OBJECT TYPE: USERAUTH

LOCATION: STLEC1 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 2-1
GROUP: N/P AUDIT TRACE REQUESTED FROM: NOT SPECIFIED
MEMBER: N/P TO: NOT SPECIFIED
SUBSYSTEM: VA1A ACTUAL FROM: 02/24/10 09:15:25.01
DB2 VERSION: V10 SCOPE: MEMBER PAGE DATE: 02/24/10

PLANNAME CONNECT TIMESTAMP TYPE DETAIL
-------- -------- ----------- -------- --------------------------------------------------------------------------------
ADMF001 WRKDDL TSO 09:15:25.01 AUTHCNTL AUTH TYPE: SYSADM
DSNTEP3 BATCH PRIV CHECKED: SECADM
AUTHID: ADMF001
SOURCE OBJECT
QUALIFIER: N/P NAME: N/P
TARGET OBJECT
QUALIFIER: N/P NAME: N/P
TEXT: GRANT DBADM TO OMVSADM
OBJECT TYPE: USERAUTH

ADMF001 WRKDDL TSO 09:15:25.01 AUTHCNTL AUTH TYPE: SYSADM
DSNTEP3 BATCH PRIV CHECKED: SECADM
AUTHID: ADMF001
SOURCE OBJECT
QUALIFIER: N/P NAME: N/P
TARGET OBJECT
QUALIFIER: N/P NAME: N/P
TEXT: GRANT DBADM TO OMVSADM
OBJECT TYPE: USERAUTH

ADMF001 WRKDDL TSO 09:15:25.01 AUTHCNTL AUTH TYPE: SYSADM
DSNTEP3 BATCH PRIV CHECKED: SECADM
AUTHID: ADMF001
SOURCE OBJECT
QUALIFIER: N/P NAME: N/P
TARGET OBJECT
QUALIFIER: N/P NAME: N/P
TEXT: GRANT DBADM TO OMVSADM
OBJECT TYPE: USERAUTH

ADMF001 WRKDDL TSO 09:15:25.01 AUTHCNTL AUTH TYPE: SYSADM
DSNTEP3 BATCH PRIV CHECKED: SECADM
AUTHID: ADMF001
SOURCE OBJECT
QUALIFIER: N/P NAME: N/P
TARGET OBJECT
QUALIFIER: N/P NAME: N/P
TEXT: GRANT DBADM TO OMVSADM
OBJECT TYPE: USERAUTH

ADMF001 WRKDDL TSO 09:15:25.01 AUTHCNTL AUTH TYPE: SYSADM
DSNTEP3 BATCH PRIV CHECKED: SECADM
AUTHID: ADMF001
SOURCE OBJECT
QUALIFIER: N/P NAME: N/P
TARGET OBJECT
QUALIFIER: N/P NAME: N/P
TEXT: GRANT DBADM TO OMVSADM
OBJECT TYPE: USERAUTH

9-4 OMEGAMON XE for DB2 PE & PM: Report Reference
**Example of a Member-Scope Audit Detail Report and Trace (Type AUTHFAIL)**

This section shows examples of a Member-Scope Audit Detail Report and Trace (Type AUTHFAIL).

**Example of a Member-Scope Audit Detail Report (Type AUTHFAIL)**

Here is an example of a Member-Scope Audit Detail report (Type AUTHFAIL).

<table>
<thead>
<tr>
<th>LOCATION: PMO2D821GANZLANG</th>
<th>GROUP: N/P</th>
<th>SUBSYSTEM: DB2</th>
<th>MEMBER: N/P</th>
<th>DB2 VERSION: V9</th>
<th>ORDER: PRIMAUTH-PLANNAME</th>
<th>AUDIT REPORT - DETAIL</th>
<th>REQUESTED FROM: NOT SPECIFIED</th>
<th>ACTUAL FROM: 12/12/03 13:04:14.30</th>
<th>TO: 12/12/03 14:21:14.30</th>
</tr>
</thead>
<tbody>
<tr>
<td>OMEGAMON XE for DB2 PE &amp; PM: Report Reference</td>
<td>9-6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PLANNAME</th>
<th>CONNECT</th>
<th>TIMESTAMP</th>
<th>TYPE</th>
<th>DETAIL</th>
</tr>
</thead>
<tbody>
<tr>
<td>JUB JUB</td>
<td>DB2CAL</td>
<td>13:04:14.30</td>
<td>AUTHFAIL</td>
<td>AUTHID CHECKED: DE#08541 PRIVILEGE: DISPLAY PROFILE</td>
</tr>
<tr>
<td>JUB 'BLANK' BA7587036AF1</td>
<td>OBJECT TYPE: ACEE</td>
<td>SOURCE OBJECT: AUDDB1</td>
<td>TARGET OBJECT: AUDTB4</td>
<td>MLS RID: MYRID</td>
</tr>
<tr>
<td>DSNRXX DB2CALL</td>
<td>SOURCE OWNER: DE#08541</td>
<td>TARGET OWNER: DE#08541</td>
<td>SECLABEL: MYSECLAB</td>
<td></td>
</tr>
<tr>
<td>JUB JUB</td>
<td>DB2CAL</td>
<td>13:05:14.30</td>
<td>AUTHFAIL</td>
<td>AUTHID CHECKED: DE#08541 PRIVILEGE: START PROFILE</td>
</tr>
<tr>
<td>JUB 'BLANK' BA7587036AF1</td>
<td>OBJECT TYPE: ACEE</td>
<td>SOURCE OBJECT: AUDDB1</td>
<td>TARGET OBJECT: AUDTB4</td>
<td>MLS RID: MYRID</td>
</tr>
<tr>
<td>DSNRXX DB2CALL</td>
<td>SOURCE OWNER: DE#08541</td>
<td>TARGET OWNER: DE#08541</td>
<td>SECLABEL: MYSECLAB</td>
<td></td>
</tr>
<tr>
<td>JUB JUB</td>
<td>DB2CAL</td>
<td>13:06:14.30</td>
<td>AUTHFAIL</td>
<td>AUTHID CHECKED: DE#08541 PRIVILEGE: STOP PROFILE</td>
</tr>
<tr>
<td>JUB 'BLANK' BA7587036AF1</td>
<td>OBJECT TYPE: ACEE</td>
<td>SOURCE OBJECT: AUDDB1</td>
<td>TARGET OBJECT: AUDTB4</td>
<td>MLS RID: MYRID</td>
</tr>
<tr>
<td>DSNRXX DB2CALL</td>
<td>SOURCE OWNER: DE#08541</td>
<td>TARGET OWNER: DE#08541</td>
<td>SECLABEL: MYSECLAB</td>
<td></td>
</tr>
<tr>
<td>JUB JUB</td>
<td>DB2CAL</td>
<td>13:07:14.30</td>
<td>AUTHFAIL</td>
<td>AUTHID CHECKED: DE#08541 PRIVILEGE: DEBUG SESSION</td>
</tr>
<tr>
<td>JUB 'BLANK' BA7587036AF1</td>
<td>OBJECT TYPE: ACEE</td>
<td>SOURCE OBJECT: AUDDB1</td>
<td>TARGET OBJECT: AUDTB4</td>
<td>MLS RID: MYRID</td>
</tr>
<tr>
<td>DSNRXX DB2CALL</td>
<td>SOURCE OWNER: DE#08541</td>
<td>TARGET OWNER: DE#08541</td>
<td>SECLABEL: MYSECLAB</td>
<td></td>
</tr>
<tr>
<td>JUB JUB</td>
<td>DB2CAL</td>
<td>13:08:14.30</td>
<td>AUTHFAIL</td>
<td>AUTHID CHECKED: DE#08541 PRIVILEGE: RENAME INDEX</td>
</tr>
<tr>
<td>JUB 'BLANK' BA7587036AF1</td>
<td>OBJECT TYPE: ACEE</td>
<td>SOURCE OBJECT: AUDDB1</td>
<td>TARGET OBJECT: AUDTB4</td>
<td>MLS RID: MYRID</td>
</tr>
<tr>
<td>DSNRXX DB2CALL</td>
<td>SOURCE OWNER: DE#08541</td>
<td>TARGET OWNER: DE#08541</td>
<td>SECLABEL: MYSECLAB</td>
<td></td>
</tr>
<tr>
<td>JUB 'BLANK' BA7587036AF1</td>
<td>OBJECT TYPE: ACEE</td>
<td>SOURCE OBJECT: AUDDB1</td>
<td>TARGET OBJECT: AUDTB4</td>
<td>MLS RID: MYRID</td>
</tr>
<tr>
<td>DSNRXX DB2CALL</td>
<td>SOURCE OWNER: DE#08541</td>
<td>TARGET OWNER: DE#08541</td>
<td>SECLABEL: MYSECLAB</td>
<td></td>
</tr>
</tbody>
</table>

LOCATION: PMO2D821GANZLANG | GROUP: N/P | SUBSYSTEM: DB2 | MEMBER: N/P | DB2 VERSION: V9 | ORDER: PRIMAUTH-PLANNAME | AUDIT REPORT - DETAIL | REQUESTED FROM: NOT SPECIFIED | ACTUAL FROM: 12/12/03 13:04:14.30 | TO: 12/12/03 14:21:14.30 |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>OMEGAMON XE for DB2 PE &amp; PM: Report Reference</td>
<td>9-6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Example of a Member-Scope Audit Trace (Type AUTHFAIL)

Here is an example of a Member-Scope Audit Trace (Type AUTHFAIL).

LOCATION: PM20DB21GANDZLANG OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-1
GROUP: N/P MEMBER: N/P SUBSYSTEM: DB2 DB2 VERSION: V9

<table>
<thead>
<tr>
<th>TIMESTAMP</th>
<th>TYPE</th>
<th>DETAIL</th>
</tr>
</thead>
<tbody>
<tr>
<td>13:04:14.30</td>
<td>AUTHFAIL</td>
<td>AUTHID CHECKED: DE#08541 PRIVILEGE: DISPLAY PROFILE</td>
</tr>
<tr>
<td>13:05:14.30</td>
<td>AUTHFAIL</td>
<td>AUTHID CHECKED: DE#08541 PRIVILEGE: START PROFILE</td>
</tr>
<tr>
<td>13:06:14.30</td>
<td>AUTHFAIL</td>
<td>AUTHID CHECKED: DE#08541 PRIVILEGE: STOP PROFILE</td>
</tr>
<tr>
<td>13:08:14.30</td>
<td>AUTHFAIL</td>
<td>AUTHID CHECKED: DE#08541 PRIVILEGE: RENAME INDEX</td>
</tr>
</tbody>
</table>

9-8 OMEGAMON XE for DB2 PE & PM: Report Reference
<table>
<thead>
<tr>
<th>PLANNAME</th>
<th>CONNECT</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>TIMESTAMP</th>
<th>TYPE</th>
<th>DETAIL</th>
</tr>
</thead>
</table>
LOCATION: PM02D821GANZLANG

PRIMAUTH CORNAME CORRMNR INSTANCE

PLANNAME CONNECT TIMESTAMP TYPE DETAIL

---

SOURCE OBJECT: AUDDB1 SOURCE OWNER: DE#08541
TARGET OBJECT: AUDTB4 TARGET OWNER: DE#08541
MLS RID : MYRID SECLABEL: MYSECLAB
TEXT: CREATE TABLE AUDTB4 (IDCOLUMN ROWID GENERATED ALWAYS NOT NULL, BELIEBIG CHAR(50)) IN AUDDB1.AUDTS1

JUB 'BLANK' BA7587036AF1 OBJECT TYPE : ACEE REASON: 0 RC: - 1

SOURCE OBJECT: AUDDB1 SOURCE OWNER: DE#08541
TARGET OBJECT: AUDTB4 TARGET OWNER: DE#08541
MLS RID : MYRID SECLABEL: MYSECLAB
TEXT: CREATE TABLE AUDTB4 (IDCOLUMN ROWID GENERATED ALWAYS NOT NULL, BELIEBIG CHAR(50)) IN AUDDB1.AUDTS1

JUB 'BLANK' BA7587036AF1 OBJECT TYPE : ACEE REASON: 0 RC: - 1

SOURCE OBJECT: AUDDB1 SOURCE OWNER: DE#08541
TARGET OBJECT: AUDTB4 TARGET OWNER: DE#08541
MLS RID : MYRID SECLABEL: MYSECLAB
TEXT: CREATE TABLE AUDTB4 (IDCOLUMN ROWID GENERATED ALWAYS NOT NULL, BELIEBIG CHAR(50)) IN AUDDB1.AUDTS1

JUB 'BLANK' BA7587036AF1 OBJECT TYPE : ACEE REASON: 0 RC: - 1

JUB JUB DB2CALL 14:21:14.30 AUTHFAIL AUTHID CHECKED: DE#08541 PRIVILEGE: X'03DB'
SOURCE OBJECT: AUDDB1 SOURCE OWNER: DE#08541
TARGET OBJECT: AUDTB4 TARGET OWNER: DE#08541
MLS RID : MYRID SECLABEL: MYSECLAB
TEXT: CREATE TABLE AUDTB4 (IDCOLUMN ROWID GENERATED ALWAYS NOT NULL, BELIEBIG CHAR(50)) IN AUDDB1.AUDTS1

JUB 'BLANK' BA7587036AF1 OBJECT TYPE : ACEE REASON: 0 RC: - 1

AUDIT TRACE COMPLETE
Field Descriptions (Audit Detail Report and Audit Trace)

This topic contains the description of the columns of the audit detail report and audit trace.

**OMEGAMON XE for DB2 PE identifiers**

The identifiers define the order of the Audit data reported. If the requester location differs from the local location, the report or trace shows REQLOC together with the appropriate name under the first column of identifiers.

The member name (MEMBER) is printed if you requested a group-scope report or trace.

**TIMESTAMP**

The time of the event.

**TYPE**

The type of event being reported. You can control which of the events is reported using TYPE with INCLUDE or EXCLUDE. Possible values are:

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTHCHG</td>
<td>Authorization change.</td>
</tr>
<tr>
<td>AUTHCNTL</td>
<td>Authorization control.</td>
</tr>
<tr>
<td>AUTHFAIL</td>
<td>Authorization failure.</td>
</tr>
<tr>
<td>BIND</td>
<td>Audited DML at bind access.</td>
</tr>
<tr>
<td>DDL</td>
<td>Audited DDL access.</td>
</tr>
<tr>
<td>DML</td>
<td>Audited DML access.</td>
</tr>
<tr>
<td>UTILITY</td>
<td>Audited utility access.</td>
</tr>
</tbody>
</table>

**DETAIL**

Each event has its own specific detail.

Your selection for the TYPE option determines which of the fields described on the following pages is printed.
Audit Report - Authorization Change Detail (Type AUTHCHG)

Authorization Change Detail (Type AUTHCHG)

The following sections list the various types of authorization changes that can be printed. They describe the fields if you select AUTHCHG. The types are sorted by IFCID.

- Set Current SQLID (IFCID 055)” on page 9-13
  This topic provides a sample and the field description of the Authorization Change type of Set Current SQLID (IFCID 055).

- “End of Identify (IFCID 083)” on page 9-14
  This topic provides a sample and the field description of the Authorization Change type of End of Identify (IFCID 083).

- “End of Signon (IFCID 087)” on page 9-15
  This topic provides a sample and the field description of the Authorization Change type of End of Signon (IFCID 087).

- “Outbound DDF Translation (IFCID 169)” on page 9-16
  This topic provides a sample and the field description of the Authorization Change type of Outbound DDF Translation (IFCID 169).

- “Inbound DDF Translation (IFCID 169)” on page 9-17
  This topic provides a sample and the field description of the Authorization Change type of Inbound DDF Translation (IFCID 169).

- “Establish Trusted Context or Reuse Trusted Context (IFCID 269)” on page 9-18
  This topic provides a sample and the field description of the Authorization Change type of Establish Trusted Context or Reuse Trusted Context (IFCID 269).

- “KERBEROS or ENCRYPTED (IFCID 319)” on page 9-20
  This topic provides a sample and the field description of the Authorization Change type of KERBEROS or ENCRYPTED (IFCID 319).
Set Current SQLID (IFCID 055)

This topic provides a sample and the field description of the Authorization Change type of Set Current SQLID (IFCID 055).

Set Current SQLID (IFCID 055) - Authorization Change

This sample shows the layout for an authorization change type of Set Current SQLID (IFCID 055):

<table>
<thead>
<tr>
<th>TYPE: SET CURRENT SQLID</th>
<th>STATUS: SUCCESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>PREVIOUS SQLID: THIS IS AN EXAMPLE OF A VERY LONG PREVIOUS SQLID THAT EXCEEDS THE LINE</td>
<td></td>
</tr>
<tr>
<td>NEW SQLID: THIS IS AN EXAMPLE OF A VERY LONG NEW SQLID THAT EXCEEDS THE LINE</td>
<td></td>
</tr>
</tbody>
</table>

Field description

The fields are described in the following:

**TYPE**  The kind of authorization change or establishment: SET CURRENT SQLID.

**STATUS**  The success or failure of the attempted authorization change. Possible values are:
- SUCCESS for a successful authorization change
- FAILURE for a failed attempt

**Note:** The SQL statement is always successful if the user has SYSADM authority.

**Derivation:** DB2 field QW0055ST

**PREVIOUS SQLID**  The initial value of the SQLID before execution of the request.

**Derivation:** DB2 field QW0055OI.

**NEW SQLID**  If the command completed successfully, the new value of the SQLID is shown. If the command did not complete successfully, the value of the attempted SQLID change is shown.

**Derivation:** DB2 field QW0055NI.
Audit Report - Authorization Change Detail (Type AUTHCHG)

End of Identify (IFCID 083)

This topic provides a sample and the field description of the Authorization Change type of End of Identify (IFCID 083).

End of Identify (IFCID 083) - Authorization Change

This sample shows the layout for an authorization change type of End of Identify (IFCID 083):

<table>
<thead>
<tr>
<th>TYPE: END OF IDENTIFY</th>
<th>STATUS: SUCCESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>PREVIOUS AUTHID: KARN</td>
<td>CURRENT SQLID: KARN</td>
</tr>
<tr>
<td>SECONDARY AUTHID: DE#03704</td>
<td></td>
</tr>
</tbody>
</table>

Field description

The fields are described in the following:

**TYPE**
The kind of authorization change or establishment: END OF IDENTIFY.

**STATUS**
The success or failure of the attempted authorization change. Possible values are:

- **SUCCESS**
  The access is permitted.

- **EXITFAIL**
  The access is denied by the authorization exit.

- **FAILURE**
  The access is denied by the security authorization facility or security system.

  *Derivation: DB2 field QW0083AD*

**PREVIOUS AUTHID**
The original value of the authorization ID, as passed to the IDENTIFY or SIGNON authorization exit.

  *Derivation: DB2 field QW0083OP*

**CURRENT SQLID**
The value of the authorization ID as set by the IDENTIFY or SIGNON authorization exit.

  *Derivation: DB2 field QW0083QD*

**SECONDARY AUTHID**
Lists the secondary authorization IDs set by the IDENTIFY or SIGNON authorization exit. If no secondary authorization IDs exist, this line is not printed. Secondary authorization IDs are printed in rows of five, up to a maximum of 49 rows (245 AUTHIDs).

  *Derivation: DB2 field QW0083SA*
End of Signon (IFCID 087)

This topic provides a sample and the field description of the Authorization Change type of End of Signon (IFCID 087).

End of Signon (IFCID 087) - Authorization Change

This sample shows the layout for an authorization change type of End of Signon (IFCID 087):

```
<table>
<thead>
<tr>
<th>TYPE: END OF SIGNON</th>
<th>STATUS: SUCCESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>PREV AUTHID: XXASP07</td>
<td>NEW AUTHID: XXASP09</td>
</tr>
<tr>
<td>SECONDARY AUTHID: XXASP09 XXASP11 XXASP26</td>
<td></td>
</tr>
</tbody>
</table>
```

Field description

The fields are described in the following:

**TYPE**  The kind of authorization change or establishment: END OF SIGNON.

**STATUS**  The success or failure of the attempted access. Possible values are:

- **SUCCESS**  The access is permitted.
- **EXITFAIL**  The access was denied by the signon authorization exit.

*Derivation:* DB2 field QW0087AD

**PREV AUTHID**  The original value of the authorization ID, as passed to the IDENTIFY or SIGNON authorization exit.

*Derivation:* DB2 field QW0087OP

**NEW AUTHID**  The value of the authorization ID as set by the IDENTIFY or SIGNON authorization exit.

*Derivation:* DB2 field QW0087QD

**SECONDARY AUTHID**  Lists the secondary authorization IDs set by the IDENTIFY or SIGNON authorization exits. If no secondary authorization IDs exist, this line is not printed. Secondary authorization IDs are printed in rows of five, up to a maximum of 49 rows (245 AUTHIDs).

*Derivation:* DB2 field QW0087SA
Outbound DDF Translation (IFCID 169)

This topic provides a sample and the field description of the Authorization Change type of Outbound DDF Translation (IFCID 169).

Outbound DDF Translation (IFCID 169) - Authorization Change

This sample shows the layout for an authorization change type of Outbound DDF Translation (IFCID 169):

```
TYPE: OUTBOUND DDF TRANSLATION REMOTE LU NAME: 'BLANK'
PREVIOUS AUTHID: PSYSAUTH
NEW AUTHID: NSYSAUTH
RESPOND LOCATION: RESPONDING LOCNM

TYPE: OUTBOUND DDF TRANSLATION REMOTE LU NAME: 'BLANK'
RESPOND LOCATION: RESPONDING LOCNM
DATABASE ALIAS: THIS IS AN EXAMPLE OF A VERY LONG ALIAS NAME THAT EXCEEDS
THE OUTPUT LINE

TYPE: OUTBOUND DDF TRANSLATION PREVIOUS AUTHID: PSYSAUTH
NEW SYSAUTHID: NSYSAUTH
RESPOND LOCATION: RESPONDING LOCNM
```

Field description

The fields are described in the following:

**TYPE**  
The kind of authorization change or establishment: OUTBOUND DDF TRANSLATION.

**REMOTE LU NAME**  
The logical unit name of the DB2 subsystem.

*Derivation:* DB2 field QW0169LU

**PREVIOUS AUTHID**  
The authorization ID before translation.

*Derivation:* DB2 field QW0169AU

**NEW AUTHID**  
The new value of the authorization ID.

*Derivation:* DB2 field QW0169NE

**RESPOND LOCATION**  
The location name of the serving DB2 subsystem. For outbound translation and for inbound translation with AUTHIDs this field shows the responding location name.

*Derivation:* DB2 field QW0169LO

**DATABASE ALIAS**  
The database alias name sent to the server.

*Derivation:* DB2 field QW0169AL

**PREV. SYSAUTHID**  
The system authorization ID before translation.

*Derivation:* DB2 field QW0169AU
Audit Report - Authorization Change Detail (Type AUTHCHG)

Inbound DDF Translation (IFCID 169)

This topic provides a sample and the field description of the Authorization Change type of Inbound DDF Translation (IFCID 169).

Inbound DDF Translation (IFCID 169) - Authorization Change

This sample shows the layout for an authorization change type of Inbound DDF Translation (IFCID 169):

<table>
<thead>
<tr>
<th>TYPE:</th>
<th>INBOUND DDF TRANSLATION REMOTE LU NAME: 'BLANK'</th>
</tr>
</thead>
<tbody>
<tr>
<td>PREVIOUS AUTHID:</td>
<td>PSYSAUTH</td>
</tr>
<tr>
<td>NEW AUTHID:</td>
<td>NSYSAUTH</td>
</tr>
<tr>
<td>RESPOND LOCATION:</td>
<td>RESPONDING</td>
</tr>
<tr>
<td>TYPE:</td>
<td>INBOUND DDF TRANSLATION REMOTE LU NAME: 'BLANK'</td>
</tr>
<tr>
<td>LOCAL LOCATION:</td>
<td>RESPONDING LOCNM</td>
</tr>
<tr>
<td>LOCATION ALIAS:</td>
<td>THIS IS AN EXAMPLE OF A VERY LONG LOCATION ALIAS NAME THAT EXCEEDS THE OUTPUT LINE</td>
</tr>
</tbody>
</table>

Field description

The fields are described in the following:

**TYPE**
The kind of authorization change or establishment: INBOUND DDF TRANSLATION.

**REMOTE LU NAME**
The logical unit name of the DB2 subsystem.

*Derivation: DB2 field QW0169LU*

**PREVIOUS AUTHID**
The authorization ID before translation.

*Derivation: DB2 field QW0169AU*

**NEW AUTHID**
The new value of the authorization ID.

*Derivation: DB2 field QW0169NE*

**RESPOND LOCATION**
The location name of the serving DB2 subsystem. For outbound translation and for inbound translation with AUTHIDs this field shows the responding location name.

*Derivation: DB2 field QW0169LO*

**LOCAL LOCATION**
The location name of the serving DB2 subsystem. For inbound translation with location alias name this field shows the local location name.

*Derivation: DB2 field QW0169LO*

**LOCATION ALIAS**
For translation type inbound this field shows the location alias name received from the requester.

*Derivation: DB2 field QW0169AL*
Establish Trusted Context or Reuse Trusted Context (IFCID 269)

This topic provides a sample and the field description of the Authorization Change type of Establish Trusted Context or Reuse Trusted Context (IFCID 269).

Establish Trusted Context or Reuse Trusted Context (IFCID 269) - Authorization Change

This sample shows the layout for an authorization change type of Establish Trusted Context or Reuse Trusted Context (IFCID 269):

<table>
<thead>
<tr>
<th>TYPE</th>
<th>ESTABLISH TRUSTED CONTEXT</th>
<th>STATUS</th>
<th>SUCCESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>OBJECT OWNER</td>
<td>ROLE</td>
<td>SQLCODE:</td>
<td>100</td>
</tr>
<tr>
<td>SECURITY LABEL</td>
<td>SECLABEL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CONTEXT NAME</td>
<td>THIS IS AN EXAMPLE OF A VERY LONG XXXXXXXXXXXXXXXX NAME THAT EXCEEDS THE OUTPUT LINE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CONTEXT ROLE</td>
<td>THIS IS AN EXAMPLE OF A VERY LONG CONTEXT ROLE THAT EXCEEDS THE OUTPUT LINE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>USER ROLE</td>
<td>THIS IS AN EXAMPLE OF A VERY LONG USER ROLE THAT EXCEEDS THE OUTPUT LINE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PREV. SYSAUTHID</td>
<td>THIS IS AN EXAMPLE OF A VERY LONG SYSTEM AUTHENTICATION ID THAT EXCEEDS THE OUTPUT LINE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>REUSE AUTHID</td>
<td>THIS IS AN EXAMPLE OF A VERY LONG REUSE AUTHENTICATION ID THAT EXCEEDS THE OUTPUT LINE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SERVAUTH NAME</td>
<td>THIS IS AN EXAMPLE OF A VERY LONG SERVER AUTHENTICATION NAME THAT EXCEEDS THE OUTPUT LINE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JOB NAME</td>
<td>THIS IS AN EXAMPLE OF A VERY LONG LOCAL JOB NAME THAT EXCEEDS THE OUTPUT LINE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ENCRYPTION</td>
<td>THIS IS AN EXAMPLE OF A VERY LONG ENCRYPTION VALUE THAT EXCEEDS THE OUTPUT LINE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TCP/IP USED</td>
<td>THIS IS AN EXAMPLE OF A VERY LONG USED TCP/IP ADDRESS THAT EXCEEDS THE OUTPUT LINE</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Field description

The fields are described in the following:

**TYPE**  
The kind of authorization change or establishment: ESTABLISH TRUSTED CONTEXT.

**STATUS**  
The status of the trusted connection:

SUCCESS  
If a trusted connection was established or reused successfully.

FAILED or FAILURE  
If a trusted connection failed, when it was tried to be established or reused.

If the status is neither SUCCESS nor FAILURE, the value itself is shown.

Derivation: DB2 field QW0269ST

**OBJECT OWNER**  
The owner of objects created in the trusted context.

**SQLCODE**  
The SQLCODE returned after executing the SQL statement.

Derivation: DB2 field QW0269SQ

**SECURITY LABEL**  
The security label.

Derivation: DB2 field QW0269SL

**CONTEXT NAME**  
The trusted context name.
Audit Report - Authorization Change Detail (Type AUTHCHG)

*Derivation: DB2 field QW0269TC*

**CONTEXT ROLE**
The default role associated with the context.
*Derivation: DB2 field QW0269RC*

**USER ROLE**
The user role.
*Derivation: DB2 field QW0269RU*

**PREV. SYSAUTHID**
The system authorization ID that is used to establish the trusted connection.
*Derivation: DB2 field QW0269SA*

**REUSE AUTHID**
The authorization ID under which a trusted connection is reused.
*Derivation: DB2 field QW0269RA*

**SERVAUTH NAME**
The SERVAUTH name of the TCP/IP security zone.
*Derivation: DB2 field QW0269SR*

**JOB NAME**
The job name for a local application.
*Derivation: DB2 field QW0269JN*

**ENCRYPTION**
The encryption value.
*Derivation: DB2 field QW0269EC*

**TCP/IP USED**
The actual communication TCP/IP address used for connection.
*Derivation: DB2 field QW0269AD*
Audit Report - Authorization Change Detail (Type AUTHCHG)

KERBEROS or ENCRYPTED (IFCID 319)

This topic provides a sample and the field description of the Authorization Change type of KERBEROS or ENCRYPTED (IFCID 319).

KERBEROS or ENCRYPTED (IFCID 319) - Authorization Change

This sample shows the layout for an authorization change type of KERBEROS or ENCRYPTED (IFCID 319):

```
TYPE: KERBEROS COMMS ADDR TYPE: TCP/IP
IP ADDR: 000102030405060718191A1B1C1D1E1F PORT: 1234
PRINCIPAL NAME: THIS IS AN EXAMPLE OF A VERY LONG REQUESTING KERBEROS NAME THAT EXCEEDS THE OUTPUT LINE

TYPE: KERBEROS COMMS ADDR TYPE: SNA
LU NAME: LUN>=V9
PRINCIPAL NAME: THIS IS AN EXAMPLE OF A VERY LONG REQUESTING KERBEROS NAME THAT EXCEEDS THE OUTPUT LINE

TYPE: ENCRYPTED COMMS ADDR TYPE: TCP/IP
IP ADDR: 000102030405060718191A1B1C1D1E1F PORT: 1234
SECURITY MECHANISM: UID Encrypt PW.

TYPE: ENCRYPTED COMMS ADDR TYPE: SNA
LU NAME: LUN>=V9
SECURITY MECHANISM: UID Encrypt PW.
```

Field description

The fields are described in the following:

**TYPE**  The type of security identity: KERBEROS or ENCRYPTED.

*Derivation*: DB2 field QW0319TY

**COMMS ADDR TYPE**

Type of communication address: SNA or TCP/IP.

*Derivation*: DB2 field QW0319CT

**IP ADDR**

If the type of the communication address is TCP/IP, it is the 16 byte hexadecimal (HLHLHLHLHLHLHLHLHLHLHLHLHLHLHLHLHLHLHLHLHL) IP address of the internal 128 bit format, where:

- \( H \) represents the high order half byte value
- \( L \) represents the low order half byte value

*Derivation*: DB2 field QW0319IPA

**PORT**  The internal port format in case of communication address type TCP/IP.

*Derivation*: DB2 field QW0319PRT

**DERIVED LOCAL UID**

Local user ID mapped by DB2.

*Derivation*: DB2 field QW0319US

**CLIENT PRODUCT ID**

The identification of the client product.

*Derivation*: DB2 field QW0319CP
Audit Report - Authorization Change Detail (Type AUTHCHG)

PRINCIPAL NAME
The requesting principal name. This can be up to 256 characters and can contain lowercase characters.

Derivation: DB2 field QW0319D1

LU NAME
If the type of the communication address is SNA, it is the 8 byte logical unit name.

Derivation: DB2 field QW0319LUN

SECURITY MECHANISM
The security mechanism. Possible values are:

- UID Encrypt PW
- Encrypt UID PW
- Encrypt UID PW NewPW
- Encrypt UID Data
- Encrypt UID PW Data
- Encrypt UID PW NewPW
- Data Encrypt UID only

Derivation: DB2 field QW0319SM
Authorization Control Detail (Type AUTHCNTL)

This topic shows detailed information about “Authorization Control Detail (Type AUTHCNTL)”.

When you select AUTHCNTL, the data is retrieved from IFCID 141 or 361, and the following fields are printed:

For the GRANTOR (IFCID 141)

<table>
<thead>
<tr>
<th>GRANTOR</th>
<th>JOB</th>
<th>OWNER TYPE</th>
<th>ROLE</th>
<th>REASON</th>
<th>SQLCODE</th>
<th>SQLCODE</th>
</tr>
</thead>
<tbody>
<tr>
<td>JUB</td>
<td></td>
<td>PRIM/SECOND AUTHID</td>
<td></td>
<td>N/P</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

OBJECT TYPE: TAB/VIEW

TEXT: GRANT INSERT ON TABLE JUB.AUDTB1 TO PUBLIC

**GRANTOR or REVOKE**
- The authorization ID of the user who issued the GRANT or REVOKE.
- This field is blank if the BY clause is used in a REVOKE statement.

**OWNER TYPE**
- The authorization type of the owner. Possible values are:
  - ROLE: A role is used.
  - PRIM/SECOND AUTHID: The user ID of the primary or the secondary authorization ID is used.
  - N/P: Not present. A blank is shown in the performance database.

**REASON**
- The reason why access was granted.
- In the Audit report set this field is only valid for GRANTS. It indicates the authorization level of the grantor. For REVOKEs and unsuccessful GRANTS, N/A is printed.
- Possible values are:
  - PACKADMA (abbreviation for PACKADM ON ALL COLLECTIONS)
  - DBCTRL
  - DBADM
  - SECADM
  - ACCCTRL (abbreviation for ACCESSCTRL)
  - SYSCTRL
  - DBMAINT
  - SYSOPR
  - PACKADMS (abbreviation for PACKADM ON A SPECIFIC COLLECTION-ID)
  - SYSADM

**SQLCODE**
- The SQL return code from the GRANT or REVOKE operation.

**OBJECT TYPE**
- The DB2 object type. Possible values are:
  - BUFFER: Buffer Pool
  - COLLECT: Collection
Audit Report - Authorization Control Detail (Type AUTHCNTL)

DATABASE
Database

DISTTYPE
Distinct Type

FUNCTION
Function

PACKAGE
Package

SCHEMA
Schema

PROCEDUR
Procedure

APPLPLAN
Application Plan

LOBTS
Large objects table space

STOGROUP
Storage Group

TAB/VIEW
Table or View

USERAUTH
System privileges, such as SYSADM or SYSOPR

SEQUENCE
Sequence

ACEE
Access control environment element

ROW
Row

TEXT
The SQL statement text associated with the GRANT or REVOKE. Long SQL text can be truncated.

For the authorization ID (AUTHID) (IFCID 361)

AUTHCNTL AUTH TYPE: SYSADM
PRIV CHECKED: EXECUTE
AUTHID: SYSADM
SOURCE OBJECT QUALIFIER: DSNTEP3
NAME: DSNTEP3
TARGET OBJECT QUALIFIER: N/P
NAME: N/P
OTHER OBJECT NAME: N/P
TEXT: N/P

Or ROLE instead of AUTHID:

AUTH TYPE: xxxxxxxxxxxxxxx
PRIV CHECKED: xxxxxxxxxxxxxxx
ROLE: xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
Audit Report - Authorization Control Detail (Type AUTHCNTL)

**AUTH TYPE**

The authority type.
Possible values are:
- SYSDBADM (System DBADM)
- DBCTRL
- DBADM
- SECADM
- ACCSCTRL (ACCESSCTRL)
- SYSADMI (Installation SYSADM)
- SQLADM
- SYSCTRL
- DBMAINT
- SYSOPR
- PACKADM
- SYSOPRI (Installation SYSOPR)
- SYSADM
- DATAACCS (DATAACCESS)
- USER

**PRIV CHECKED**

The privilege that was checked. Possible values are provided in the DB2 macro DSNDQW05.

**OBJECT TYPE**

The DB2 object type.
Possible values are:
- ACEE
- BUFFER (Bufferpool)
- COLLECT (Collection)
- DATABASE
- DISTTYPE (Distinct Type)
- FUNCTION
- SESSIONV (Session Variable)
- JAR
- PACKAGE
- ROLE
- SCHEMA
- TRUSTCTX (Trusted Context)
- PROCEDUR (Procedure)
- APPLPLAN (Application Plan)
- LOBTS (LOB Tablespace)
- STOGROUP (Storage Group)
- TAB/VIEW (Table or View)
Audit Report - Authorization Control Detail (Type AUTHCNTL)

- USERAUTH (User Auth)
- SEQUENCE
- ROW

AUTHID or ROLE
The authorization ID or the role that has the authority.

SOURCE OBJECT - QUALIFIER
The source object qualifier or owner.

SOURCE OBJECT - NAME
The source object name.

TARGET OBJECT - QUALIFIER
The target object qualifier or owner.

TARGET OBJECT - NAME
The target object name.

OTHER OBJECT - NAME
The other object name or subsystem parameter.

TEXT
The SQL statement (truncated at 4000 bytes).
Audit Report - Authorization Failure Detail (Type AUTHFAIL)

Authorization Failure Detail (Type AUTHFAIL)

This topic shows detailed information about “Authorization Failure Detail (Type AUTHFAIL)”.

When you select AUTHFAIL, the data is retrieved from IFCID 140, and the following fields are printed:

```
AUTHID CHECKED: DE#08541 PRIVILEGE: VALID. SECLABEL
OBJECT TYPE : ACEE REASON: 0 RC: 0
SOURCE OBJECT : AUDDB1 SOURCE OWNER: DE#08541
TARGET OBJECT : AUDTB4 TARGET OWNER: DE#08541
MLS RID : XXXXX SECLABEL: XXXXXXXX
TEXT: CREATE TABLE AUDTB4 (IDCOLUMN ROWID GENERATED ALWAYS NOT
NULL, BELIEBIG CHAR(50)) IN AUDDB1.AUDTS1
```

Field description

Here is a description of the field labels shown in the previous example:

**AUTHID CHECKED**
The authorization ID causing the failure.

**PRIVILEGE**
The privilege that was checked. Possible values are provided in the DB2 macro DSNDQW02.

**OBJECT TYPE**
The DB2 object type. Possible values are:
- **ACEE** Access control environment element (ACEE)
- **APPLPLAN** Application Plan
- **BUFFER** Buffer Pool
- **COLLECT** Collection
- **DATABASE** Database
- **DISTTYPE** Distinct Type
- **FUNCTION** Function
- **LOBTS** Table Space
- **PACKAGE** Package
- **PROCEDUR** Procedure
- **ROW** Row
- **SCHEMA** Schema
- **SEQUENCE** Sequence
Audit Report - Authorization Failure Detail (Type AUTHFAIL)

**STOGROUP**
Storage Group

**TAB/VIEW**
Table or View

**USERAUTH**
System privileges, such as SYSADM or SYSOPR

**REASON**
The user-defined reason code from the access control authorization exit routine.

**RC**
The return code from the access control authorization exit routine. A value of 0 means “not applicable”.

**SOURCE OBJECT**
If the OBJECT TYPE field is not blank, this field displays the name of the object against which the authorization was checked.

If the object type is blank, then this field displays the name of the object being created. It is valid only when the privilege is CREATE ALIAS, CREATE DBA, CREATE DBC, or CREATE STOGROUP.

Otherwise, N/A is printed.

**SOURCE OWNER**
If the OBJECT TYPE field is not blank, this field displays the qualifier of the object against which the authorization was checked. It is valid only for qualifiable objects.

If the object type is blank, this field displays the qualifier of the alias being created. It is valid only when the privilege is CREATE ALIAS.

Otherwise, N/A is printed.

**TARGET OBJECT**
The name of the object being defined. It is valid only when the target owner field is valid. Otherwise, N/A is printed.

**TARGET OWNER**
The qualifier of the object being defined. It is valid when the privilege is CREATE INDEX or CREATE TABLE. It is also valid for a CREATE VIEW authorization check against the set of CREATE VIEW, SELECT, INSERT, DELETE, and UPDATE privileges. Otherwise, N/A is printed.

**MLS RID**
The record identifier (RID) of the Multilevel Security (MLS) table that is updated or deleted.

**SECLABEL**
The security label of a row in the Multilevel Security (MLS) table.

**TEXT**
The SQL statement text associated with the failure. Long SQL statement text can be truncated, depending on the amount of space available.
This topic shows detailed information about “DML at Bind Access Detail (Type BIND)”.

When you select the BIND type, the data is retrieved from IFCID 145, and the following fields are printed:

<table>
<thead>
<tr>
<th>TYPE DETAIL</th>
</tr>
</thead>
<tbody>
<tr>
<td>BIND PACKAGE: PMDIB11.DSNREXX.DSNREXX.X'174B9CF31C56B7C2'</td>
</tr>
<tr>
<td>TYPE: INSERT STMT# 0 ISOLATION(CS) KEEP UPD LOCKS: NO</td>
</tr>
<tr>
<td>TEXT: INSERT INTO PRL.AUDTB1 OVERRIDING USER VALUE VALUES(NULL, 'HEID1', 'AXEL', 1)</td>
</tr>
<tr>
<td>DATABASE: 264 TABLE OBID: 3 STMT ID: 0</td>
</tr>
<tr>
<td>ACCESS CTRL SCHEMA: N/P</td>
</tr>
<tr>
<td>ACCESS CTRL OBJECT: N/P</td>
</tr>
</tbody>
</table>

**Field description**

**PACKAGE/DBRM NAME**

The name of the database request module (DBRM) or package containing the DML statement being bound.

A package name is made up of the following parts:

**Location**

The location name is applicable only to packages, otherwise 'BLANK' is printed.

**Collection ID**

The package collection ID is applicable only to packages, otherwise 'BLANK' is printed.

**Package ID**

The program name for DBRMs or the package ID for packages.

**Consistency token**

A hexadecimal dump of the DB2 timestamp of the program during precompilation. This field contains the value in the TIMESTAMP column of SYSIBM.SYSDBRM. The value represents the time of the precompilation in internal format, that is, modified store clock (STCK) format.

**TYPE**

The type of statement being bound.

**STMT#**

The statement number in the program or DBRM involved in the bind.

**ISOLATION**

The isolation of the bind. Possible values are:

- CS  Cursor stability
- RR  Repeatable read
- RS  Read stability
- UR  Uncommitted read

**KEEP UPD LOCKS**

Indicates if an update lock is kept. YES is only valid if the value in the TYPE field is SQL OPEN CURSOR and the value in the ISOLATION field is RR or RS.

**TEXT**

The SQL statement text associated with the BIND. If SQL text is not present, N/P is printed. Long SQL text can be truncated.
Audit Report - DML at Bind Access Detail (Type BIND)

DATABASE
Either the name or the decimal DBID of the database that contains the
auditable DB2 table. The name is printed if known, otherwise
OMEGAMON XE for DB2 PE reports a decimal DBID.

TABLE OBID
The object identifier (OBID) of the DB2 table.

Note: DATABASE and TABLE OBID are repeating fields and can have
more than one entry in the report or trace. Two or more occurrences result
from a DML statement that refers to two or more DB2 tables.

STMT ID
The SQL unique statement ID.

ACCESS CTRL SCHEMA
The name of the access control schema.

ACCESS CTRL OBJECT
The name of the access control object.
DDL Access Detail (Type DDL)

This topic shows the various types of DDL Access detail that can be printed. It describes the corresponding fields if you select DDL. The types are sorted by IFCID.

"Create Trusted Context or Alter Trusted Context (IFCID 270)" on page 9-31
This topic shows a sample and the field description of a DDL change type of Create Trusted Context or Alter Trusted Context (IFCID 270).

"DDL data retrieved from IFCID 142 or 271" on page 9-32
This topic shows a sample and the field description of DDL data retrieved from IFCID 142 or 271.
Create Trusted Context or Alter Trusted Context (IFCID 270)

This topic shows a sample and the field description of a DDL change type of Create Trusted Context or Alter Trusted Context (IFCID 270).

The following sample shows the layout for an DDL change type of Create Trusted Context or Alter Trusted Context (IFCID 270):

<table>
<thead>
<tr>
<th>TYPE: CREATE TRUSTED CONTEXT</th>
<th>SQLCODE: 100</th>
</tr>
</thead>
<tbody>
<tr>
<td>TEXT: THIS IS THE VARIABLE LENGTH SQL STATEMENT WHEN A TRUSTED CONTEXT IS CREATED OR ALTERED. MAXIMUM LENGTH IS 4000.</td>
<td></td>
</tr>
</tbody>
</table>

Field description

The fields are described in the following:

**TYPE**  The type of trusted context. Possible values are:

CREATE TRUSTED CONTEXT or CREATE
If a trusted context is created.

ALTER TRUSTED CONTEXT or ALTER
If a trusted context is altered.

*Derivation:* DB2 field QW0270TY

**SQLCODE**  The SQL return code from the CREATE or ALTER TRUSTED CONTEXT statement.

*Derivation:* DB2 field QW0270SQ

**TEXT**  The SQL statement (truncated at 4000 bytes).

*Derivation:* DB2 field QW0270SS
DDL data retrieved from IFCID 142 or 271

This topic shows a sample and the field description of DDL data retrieved from IFCID 142 or 271.

If the data is retrieved from IFCID 142, ...

If you select DDL and the data is retrieved from IFCID 142 and the following fields are printed:

TABLE NAME: AUDTB1  OWNER: JUB  CREATOR: JUB
DATABASE: 274  TABLE OBID: 3  TYPE: CREATE
OWNER TYPE: PRIM/SECOND AUTHID  SECLABEL: XXXXXXXX  MLS: XXXX
R/C ACCESS CTRL: COLUMN
TEXT: CREATE TABLE JUB.AUDTB1 (IDCOLUMN INTEGER GENERATED ALWAYS AS IDENTITY, NNAME VARCHAR(50) NOT NULL, VNAME CHAR(10) NOT NULL, ANZAHL INTEGER NOT NULL) AUDIT ALL IN AUDDB1.AUDTS1

**TABLE NAME**
The name of the audited DB2 table.

**OWNER**
The authorization ID of the owner of the audited DB2 table.

**CREATOR**
The authorization ID of the creator of the DB2 table.

**DATABASE**
Either the name or the decimal DBID of the database that contains the auditable DB2 table. The name is printed if known, otherwise OMEGAMON XE for DB2 PE reports a decimal DBID.

**TABLE OBID**
The object identifier (OBID) of the auditable table associated with the access.

**TYPE**
The statement type. Possible values are:
- CREATE
- DROP
- ALTER

**OWNER TYPE**
The type of the table owner. Possible values are:
- ROLE A role is used.
- PRIM/SECOND AUTHID
  The user ID of the primary or the secondary authorization ID is used.
- N/P A blank is shown in the performance database.
- N/A A blank is shown in the performance database.

**SECLABEL**
The security label of the user.

**MLS**
The Multilevel Security (MLS) table can contain the following values:
- Y For a Create or Drop operation of a table that has multilevel security, or for an Alter operation of a table to add a security label column.
- N For an Alter operation of a table that has multilevel security.
- NONE The table does not have multilevel security.
Audit Report - DDL Access Detail (Type DDL)

N/P Not present. A blank is shown in the performance database.

N/A A blank is shown in the performance database.

R/C ACCESS CTRL
The access control field contains data about ROW-LEVEL and COLUMN-LEVEL (R/C) ACCESS CONTROL in DDL. It can have the following values:

'R' (ROW)
Activates row-level access control.

'C' (COLUMN)
Activates column-level access control.

'B' (BOTH)
Activates row-level and column-level access control.

' ' (NO)
Activates no access control.

TEXT The SQL statement text associated with the table access. Long SQL text can be truncated.

If the data is retrieved from IFCID 271, ...

If you select DDL and the data is retrieved from IFCID 271, the following fields are printed:

<table>
<thead>
<tr>
<th>TYPE</th>
<th>DETAIL</th>
</tr>
</thead>
<tbody>
<tr>
<td>DDL</td>
<td>CHANGE TYPE: CREATE OBJECT: ROW PERMISSION SQLCODE: XXXX</td>
</tr>
<tr>
<td></td>
<td>TEXT: xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx</td>
</tr>
</tbody>
</table>

CHANGE TYPE
Identifies the SQL statement type:

CREATE or C
Creates row permission or column mask.

DROP or D
Drops row permission or column mask.

ALTER or A
Alters row permission or column mask.

Otherwise, a hexadecimal value is shown.

OBJECT
Identifies the object type:

• Row permission (R)
• Column mask (M)

Otherwise, a hexadecimal value is shown.

SQLCODE
The SQL code from the execution of the CREATE, DROP, or ALTER statement.

TEXT The SQL statement text associated with the table access. The maximum length is 4000 bytes. Long SQL text can be truncated.
DML Access Detail (Type DML)

This topic shows a sample and the field description of “DML Access Detail (Type DML)”. When you select DML, the data is retrieved from IFCID 143 and 144, and the following fields are printed:

<table>
<thead>
<tr>
<th>TYPE DETAIL</th>
</tr>
</thead>
<tbody>
<tr>
<td>DML TYPE : 1ST WRITE</td>
</tr>
<tr>
<td>DATABASE: 31B</td>
</tr>
<tr>
<td>PAGESET : 41</td>
</tr>
<tr>
<td>STMT ID : 0</td>
</tr>
<tr>
<td>TABLE OBID: 42</td>
</tr>
<tr>
<td>LOG RBA : X’0042ECF1B144’</td>
</tr>
</tbody>
</table>

Field description

TYPE The type of access. It is determined by the IFCID (143 is a WRITE and 144 is a READ).

STMT ID The statement ID.

DATABASE Either the name or the decimal DBID of the database that contains the auditable DB2 table. The name is printed if known, otherwise OMEGAMON XE for DB2 PE reports a decimal DBID.

TABLE OBID The object identifier (OBID) of the auditable DB2 table associated with the access.

PAGESET Either the name or the decimal PSID of the page set that contains the auditable DB2 table. The name is printed if known, otherwise OMEGAMON XE for DB2 PE reports a decimal PSID. If neither field is available, N/A is printed in this field.

LOG RBA The log relative byte address of the current unit of recovery. It is printed in hexadecimal, when present.
Utility Access Detail (Type UTILITY)

This topic shows a sample and the field description of the “Utility Access Detail (Type UTILITY)”.

When you select UTILITY, the data is retrieved from IFCID 024, and the following fields are printed:

- **NAME**: LOAD
- **ID**: DSNTEX
- **DATABASE**: DBASE1
- **PHASE**: RELOAD
- **PAGESET**: PSET1
- **TYPE**: RECORD

### Field description

Here is a description of the field labels shown in the previous example:

- **NAME**
  The name of the utility.

- **ID**
  The DB2 user's identification of the utility.

- **DATABASE**
  Either the name or the decimal DBID of the database that contains the auditable DB2 object.
  
  For the report entry describing the start of a utility (IFCID 023), N/P is printed. To determine the real value, find the corresponding entry describing the utility object or phase change (IFCID 024), or the entry describing the utility end information.

- **PHASE**
  The utility phase identification.

- **PAGESET**
  Either the name or the decimal PSID of the page set that contains the auditable DB2 object. If neither field is present, N/A is printed.

- **TYPE**
  The type of utility access. For the utility phase UTILINIT and UTILTERM, N/A is printed.
Audit Report - Utility Access Detail (Type UTILITY)
Chapter 10. The Audit File Data Set and Output Record

The FILE subcommand formats DB2 Audit records and writes them to sequential data sets that can be loaded into DB2 tables.

The audit FILE subcommand produces up to seven sequential variable-blocked data sets. You can use FILE to separate the various audit types by specifying one audit type per FILE subcommand.

The content of the output data set is determined by the options you specify for the FILE subcommand and by the input DB2 audit trace data processed.

Each output record is divided into several parts:

- The *Standard Header* section contains header data common to all records. This section is at the beginning of each record. It contains DB2 identifier information known as the correlation header. It also contains the DB2 distributed network header information associated with the record.
- The *Data* section lists data unique to the audit category of the record. Each audit type maps the areas of the record differently. Records which share the same header information have a repeating subtype.

File data is written to a File data set. The following types of records are created:

- Bind
- Auth Change
- Auth Control
- DDL
- Auth Failure
- SQL
- Utility

Descriptions of the layouts of these records can be found in the RKO2SAMP library under the following names:

- **DGOXDBND**
  - Bind
- **DGOXDCHG**
  - Auth Change
- **DGOXDCNT**
  - Auth Control
- **DGOXDDDL**
  - DDL
- **DGOXDDML**
  - DML
- **DGOXDFAI**
  - Auth Failure
- **DGOXDSQL**
  - SQL

© Copyright IBM Corp. 1985, 2013
Audit Report - File Data Set

DGOXDUTI
Utility
Part 4. The Explain Report

These topics provide information about the Explain report.

The Explain report of OMEGAMON XE for DB2 PE is divided into sections. The sections that are shown in the Explain report depend on:

- Which object is to be explained.
  
  Most sections are applicable to all explain functions, but there are a few which are applicable to selected functions only.

- The requested level of detail.

At normal completion of Explain, the last pages of the output show a summary of the OMEGAMON XE for DB2 PE explain execution. This is called the *Summary Report*.

**Note:** For an introduction to the Explain report set and general explain information refer to the *Reporting User's Guide*. For details on how to specify EXPLAIN commands refer to the *Report Command Reference*.

Chapter 11, “The Page Header,” on page 11-1

The explain page header is printed on every page of the Explain report.

Chapter 12, “Object Identification,” on page 12-1

Use the page header to quickly identify the object that is explained. A complete identification is contained in the object identification section of the report.

Chapter 13, “Table PLAN_TABLE Data,” on page 13-1

This section of the Explain report lists the raw EXPLAIN data as found in the DB2 table PLAN_TABLE.

Chapter 14, “Access Path Data,” on page 14-1

This section of the Explain report introduces the access path chosen by DB2.

Chapter 15, “Index Data,” on page 15-1

If an index is used, that is, an access path other than TABLE SPACE SCAN, information about this index is shown. The data is derived from the SYSIBM.SYSINDEXES table.

Chapter 16, “Key Data,” on page 16-1

If an index is used and LEVEL(INDEXES) is not specified, information about the key columns is presented. The data is derived from the SYSIBM.SYSKEYS and SYSIBM.SYSCOLUMNS tables.

Chapter 17, “Key Distribution Data,” on page 17-1

This section introduces the Key Distribution Data.

Chapter 18, “Table and Table Space Data,” on page 18-1

This section of the Explain report shows information for the accessed table and its corresponding table space.

Chapter 19, “Host Variable Data,” on page 19-1

This section of the Explain report is produced if HOSTVAR(YES) is specified for the OMEGAMON XE for DB2 PE explain plan or package.

Chapter 20, “Bind Plan Data,” on page 20-1

This section of the Explain report is shown if the object being explained is a plan or query number for a mini plan created by the bind process. Information related to the binding of the plan, such as plan binder, bind time, or isolation level, is presented.
Chapter 21, “Bind Package Data,” on page 21-1
This section of the Explain report is shown if the object being explained is a package.

Chapter 22, “Summary Report,” on page 22-1
This section introduces the summary report.

Chapter 23, “EXPLAIN PLAN Command,” on page 23-1
This section introduces the EXPLAIN PLAN command.

Chapter 24, “EXPLAIN PACKAGE Command,” on page 24-1
This section introduces the EXPLAIN PACKAGE command.

Chapter 25, “EXPLAIN SQLSTMT Command,” on page 25-1
This section shows an example of an Explain report generated with the EXPLAIN SQLSTMT command.
Chapter 11. The Page Header

The explain page header is printed on every page of the Explain report.

For details on how to specify EXPLAIN PACKAGE commands refer to the Report Command Reference.

Explain - General Page Header

This sample shows the header of an Explain report, in its general form:

ACTUAL AT: 02/18/13 15:05:08  OMEGAMON XE DB2 PE (V5.2)  PAGE : 1-1
LOCATION : PMODA11  DGO0TPG3  USER AUTHID: XRK
SUBSYSTEM: DA11  DETAIL  CURR.SQLID : XRK

Field description

The header contains the following information, described in the order left block, middle block, right block:

ACTUAL AT
The date and time at which the DB2 subsystem (specified in SUBSYSTEM) containing explain functions, is accessed to collect the requested data.

LOCATION
The location name of the DB2 subsystem specified in SUBSYSTEM.

SUBSYSTEM
The ID of the DB2 subsystem that generated the data.

OMEGAMON XE for DB2 PE (Vn)
The product name and version.

EXPLAIN
The object to be explained as specified in the EXPLAIN command (for example, QMFQUERY, PACKAGE, or PLAN).

<object type>
The name of the object to be explained as specified in the EXPLAIN command. It can have one of the following values:

SQL STATEMENT
None

QUERYNO
Query number

PLAN  Plan name, such as LARGEPLAN.

PACKAGE
Package ID

QMFQUERY
Unqualified QMF query name

<level>
The level of the report specified in the LEVEL subcommand option. Values are:

• SUMMARY
• SQL


Explain Report - Page Header

- BASIC
- INDEX
- DETAIL
- NO RAW EXPLAIN DATA
- KEY DISTRIBUTION

**PAGE** The page number in the format *lll-nnnn*, where *lll* denotes the report number and *nnnn* the page number within the report.

**DB2 VERSION**
The version and release of the DB2 subsystem specified in SUBSYSTEM.

**USER AUTHID**
The user authorization ID.

**CURR.SQLID**
The current SQLID as specified in the SQLID option, or the default.
Chapter 12. Object Identification

Use the page header to quickly identify the object that is explained. A complete identification is contained in the object identification section of the report.

“Plan Identification” on page 12-2
This section shows examples of identification plan with DBRMs and with packages.

“Package Identification” on page 12-3
This section shows examples of package identification.

“QMF Query Identification” on page 12-4
This section shows examples of QMF Query identification.

“SQL Text Identification” on page 12-5
If an SQL statement identified by its text is explained by the EXPLAIN SQLSTMT command, the object identification section shows the statement text.

“SQL Query Number Identification” on page 12-6
If an SQL statement identified by a query number is explained by the EXPLAIN QUERYNO command, there is no object identification section.
Plan Identification

This section shows examples of identification plan with DBRMs and with packages.

If a plan is explained by the EXPLAIN PLAN command, the object identification section of the report shows the following for each SQL statement:

- Plan name
- DBRM or package name
- Statement number
- Statement text
- Statement-related information

Examples are shown in "Explain Identification - Plan with DBRMs" and "Explain Identification - Plan with Packages."

**Explain Identification - Plan with DBRMs**

Here is an example of an Identification - Plan with DBRMs.

```sql
PLAN LOCATION :DSNAPC1
PLAN NAME :FVTXPLAN
DBRM NAME :CHOLD
DBRM VERSION ID :  
STATEMENT NUMBER : 182

SQL STATEMENT READ FROM SYSIBM.SYSTMT:
DECLARE C2 CURSOR FOR
   SELECT NAME, BINDDATE, BINDTIME, ISOLATION
   FROM SYSIBM.SYSPLAN
   WHERE CREATOR = USER AND NAME IN ('EEE2', 'EEE3', 'EEE4')
   ORDER BY NAME
STATUS : COMPILED-BOUND USING DEFAULTS FOR INPUT VARIABLES
ISOLATION: CURSOR STABILITY
```

**Explain Identification - Plan with Packages**

Here is an example of an Identification - Plan with packages.

```sql
PLAN LOCATION :DSNAPC1
PLAN NAME :FVTXPLAN
PACKAGE LOCATION :DSNAPC1
PACKAGE COLLECTION: MIXCOL
PACKAGE ID :MIX
PACKAGE VERSION ID: VER3
STATEMENT NUMBER : 87

SQL STATEMENT READ FROM SYSIBM.SYSPACKSTMT:
DECLARE CURSOR_1 CURSOR FOR
   SELECT EMPNO, LASTNAME, WORKDEPT, BIRTHDATE
   FROM DSN8610.EMP
   WHERE (EMPNO BETWEEN '000170' AND '000240' AND WORKDEPT IN ('D01', 'E21', 'X23')) OR (EMPNO = '000100' AND (WORKDEPT = (SELECT MIN (DEPTNO)
   FROM DSN8610.DEPT
   WHERE MGRNO = '000050') OR WORKDEPT = (SELECT MAX (DEPTNO)
   FROM DSN8610.DEPT
   WHERE ADMRDEPT = 'A00')))
   ORDER BY EMPNO
STATUS : COMPILED-BOUND USING DEFAULTS FOR INPUT VARIABLES
ISOLATION: CURSOR STABILITY
```
Package Identification

This section shows examples of package identification.

If a package is explained by the EXPLAIN PACKAGE command, the object identification section of the report shows the following for each SQL statement:

- Full package name
- Statement number
- Statement text
- Statement-related information

For details on how to specify EXPLAIN PACKAGE commands refer to the Report Command Reference.

Explain Identification - Package

Here is an example of a package identification.

```
PACKAGE LOCATION : PMODA11
PACKAGE COLLECTION: KO2EX520
PACKAGE ID : DGO@TPG3
PACKAGE VERSION ID: OMPE_FINAL
STATEMENT NUMBER : 1011071
SQL STATEMENT READ FROM SYSIBM.SYSPACKSTMT:

DECLARE C_DGOYTPG_71 CURSOR WITH HOLD FOR
SELECT LOCATION, COLLID, NAME, CONTOKEN, OWNER, CREATOR, TIMESTAMP, BINDTIME,
QUALIFIER, PKSIZE, AVGSIZE, SYSENTRIES, VALID, OPERATIVE, VALIDATE,
ISOLATION, RELEASE, EXPLAIN, QUOTE, COMMA, HOSTLANG, CHARSET, MIXED, DEC31,
DEFERPREP, SOLENR, HOSTNR, PDSNAME, DEGREE, GROUP_MEMBER, DYNAMICRULES, REOPTVAR, KEEP_DYNAMIC,
TYPE, DBPROTOCOL, FUNCTIONTS, OPTHINT, ENCODING_CCSID, IMMEDIATE_WRITE, RELBOUND,
CATENCODE, REMARKS
FROM DGO_SYSPACKAGE
WHERE LOCATION LIKE :HV_LOC71_LOCATION AND COLLID LIKE :HV_LOC71_COLLID AND
NAME LIKE :HV_LOC71_NAME AND VERSION LIKE :HV_LOC71_VERSION
ORDER BY LOCATION, COLLID, NAME, PDSNAME, DEGREE

STATUS : COMPILED-BOUND USING DEFAULTS FOR INPUT VARIABLES
ISOLATION: UNCOMMITTED READ / FROM SYSPACKAGE
```
QMF Query Identification

This section shows examples of QMF Query identification.

If a QMF query is explained by the EXPLAIN QMFQUERY command, the object identification section of the report shows the following for each SQL statement:

- Full QMF query name
- Statement text

For details on how to specify EXPLAIN QMFQUERY commands refer to the Report Command Reference.

Explain Identification - QMF Query

Here is an example of the QMF query identification.

QMFQUERY: USR1.MY_QUERY

```sql
--
--SELECT * FROM A CATALOG TABLE
--

SELECT NAME, CREATOR
FROM SYSIBM.SYSTABLES
WHERE CREATOR NOT LIKE 'SYSIBM%'
AND CREATOR LIKE 'XXASP%'
ORDER BY CREATOR, NAME
--
```
SQL Text Identification

If an SQL statement identified by its text is explained by the EXPLAIN SQLSTMT command, the object identification section shows the statement text.

For details on how to specify EXPLAIN SQLSTMT commands refer to the Report Command Reference.

Explain Identification - SQL Text

Here is an example of the SQL Text Identification.

```sql
SELECT * FROM SYSIBM.SYSPACKAGE
WHERE NAME = 'DGO@TPG3'
AND COLLID = 'KOZEKSSG'
```
SQL Query Number Identification

If an SQL statement identified by a query number is explained by the EXPLAIN QUERYNO command, there is no object identification section.

For details on how to specify EXPLAIN QUERYNO commands refer to the Report Command Reference.
Chapter 13. Table PLAN_TABLE Data

This section of the Explain report lists the raw EXPLAIN data as found in the DB2 table PLAN_TABLE.

For details on how to specify EXPLAIN commands refer to the Report Command Reference.

Explain Report - PLAN_TABLE Report Block

This is an example of the PLAN_TABLE report block.

The report field labels are the full PLAN_TABLE column names. For a detailed description of the PLAN_TABLE columns, refer to the DB2 SQL Reference. If the report field value is a long name or a long value, it is marked with an asterisk (*) at the end of the report field value, and it is reported at the end of the block in its full length.

The plan table is searched by using the bind time of the package. There might be multiple occurrences of the package with different bind times. For example, the DB2 system catalog might contain the latest package that is created by the DB2 command BIND or REBIND. It might also contain a former package version that is activated by the DB2 command REBIND SWITCH.
Chapter 14. Access Path Data

This section of the Explain report introduces the access path chosen by DB2.

For details on how to specify EXPLAIN commands refer to the Report Command Reference.

Explain Report - Access Path Data Block

The following example shows that the access path is contained in a frame.

| MATCHING INDEX SCAN WITH SCAN OF REFERENCED DATA PAGES |
| NUMBER OF MATCHING COLUMNS: 1 - THE INDEX HAS 4 COLUMNS |
| NON-CLUSTERED INDEX SCAN WILL BE USED |
| OPTIMIZER EXPECTS DYNAMIC PREFETCH |
| PAGE RANGE SCAN WILL NOT BE USED |
Chapter 15. Index Data

If an index is used, that is, an access path other than TABLE SPACE SCAN, information about this index is shown. The data is derived from the SYSIBM.SYSINDEXES table.

For details on how to specify EXPLAIN commands refer to the Report Command Reference.

Explain Report - Index Data Block

Here is an example of the index data block section.

```
INDEX: SYSIBM.DSNKKX01 -----------------------------------------------
STATTIME: 2013-02-16-14.45.54.186869
CREATED : 0001-01-01-00.00.00.000000 ALTERED: 2003-09-21-23.30.17.962937
FULL KEY CARD: 885,PAGES : 16,LEVELS: 2,CLUSTERING: Y
1ST KEY CARD: 1,SPACE : 245.760K,UNIQUE: YES,CLUSTERED: N
INDEX TYPE : 2,PGSIZE: 4096,BFPOOL: BP0,DB.NAME : DSNDB06
CLUSTERRATIO : 84.6328%,ERRULE: NO,CLRULE: NO,IXSPACE : DSNKKX01
MAX.PIECESIZE: 0,COPY : NO,COPYLRSN: X'000000000000'
```
Accounting Report
Chapter 16. Key Data

If an index is used and LEVEL(INDEXES) is not specified, information about the key columns is presented. The data is derived from the SYSIBM.SYSKEYS and SYSIBM.SYSCOLUMNS tables.

If a matching index scan is used in the access path, the report indicates the columns that are used in the index scan. This is indicated by an arrow (<<==) in the column named KEY USED. The number of arrows corresponds to the contents of the matching columns field MATCHCOLS in the PLAN_TABLE report block.

For details on how to specify EXPLAIN commands refer to the Report Command Reference.

Explain Report - Key Data Block

Here is an example of the explain key data block section.

<table>
<thead>
<tr>
<th>KEY NO.</th>
<th>COLUMN NAME</th>
<th>COL.TYPE</th>
<th>LNG</th>
<th>NULL</th>
<th>CARD. ORDER</th>
<th>TYPENAME</th>
<th>KEY USED</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>LOCATION</td>
<td>VARCHAR</td>
<td>128</td>
<td>NO</td>
<td>1 ASC.</td>
<td>SYSIBM</td>
<td>X'40404040 X'40404040 &lt;&lt;==</td>
</tr>
<tr>
<td>2</td>
<td>COLLID</td>
<td>VARCHAR</td>
<td>128</td>
<td>NO</td>
<td>87 ASC.</td>
<td>SYSIBM</td>
<td>X'4144424C X'55545255</td>
</tr>
<tr>
<td>3</td>
<td>NAME</td>
<td>VARCHAR</td>
<td>128</td>
<td>NO</td>
<td>576 ASC.</td>
<td>SYSIBM</td>
<td>X'41444232 X'535F5550</td>
</tr>
<tr>
<td>4</td>
<td>VERSION</td>
<td>VARCHAR</td>
<td>122</td>
<td>NO</td>
<td>55 ASC.</td>
<td>SYSIBM</td>
<td>X'31404040 X'56383231</td>
</tr>
</tbody>
</table>

© Copyright IBM Corp. 1985, 2013
Accounting Report
Chapter 17. Key Distribution Data

This section introduces the Key Distribution Data.

If LEVEL(KEYDIST) has been specified and RUNSTATS has produced key distribution information for the first column of the index, a section in the report shows the distribution of up to ten most frequently used key values. The data is derived from the SYSIBM.SYSTABLES.CREATOR.

For details on how to specify EXPLAIN commands refer to the Report Command Reference.

**Explain Report - Key Distribution Data Block**

This example shows the key distribution data section.

```
KEY DISTRIBUTION FOR : SYSIBM.SYSTABLES.CREATOR
SYSLIB (27%),DSN8230 (11%),U473298 ( 9%),Q ( 7%),XXASP09 ( 5%)
U01 ( 5%),USR2 ( 5%),XXASP32 ( 3%),XXASP16 ( 3%),USER001 ( 2%)
```
Accounting Report
Chapter 18. Table and Table Space Data

This section of the Explain report shows information for the accessed table and its corresponding table space.

The data is derived from the SYSIBM.SYSTABLES and SYSIBM.SYSTABLESPACE tables.

For details on how to specify EXPLAIN commands refer to the Report Command Reference.

Explain Report - Table Data Block

This is an example of the Explain Table Data section.

If table space scan has been selected as the access path method, and INDEX(NO) is not specified, a separate block is presented for each available index on the subject table along with information about key columns.

See Chapter 15, “Index Data,” on page 15-1 and Chapter 16, “Key Data,” on page 16-1 for the layout of these blocks.
Chapter 19. Host Variable Data

This section of the Explain report is produced if HOSTVAR(YES) is specified for the OMEGAMON XE for DB2 PE explain plan or package.

If you define host variables which are not consistent with the corresponding column definition, DB2 selects an inefficient access path.

In “Explain Report - Host Variables Data Block,” the access path selected is table space scan even though an index is defined on the only column referenced in the WHERE clause. As the example in “Explain Report - Host Variables Data Block” shows, DB2 has selected table space scan because the column definition is three characters, but the corresponding host variable is defined as four characters. By changing the host variable definition to three characters, a matching index scan is selected by DB2.

For details on how to specify EXPLAIN commands refer to the Report Command Reference.

Explain Report - Host Variables Data Block

Here is an example of an Explain Report for the Host Variables Data block

DECLARE C1 CURSOR FOR
SELECT DEPTNO, DEPTNAME, LOCATION
FROM DSN8610.DEPT
WHERE DEPTNO = :HOSTVAR_STRUCTURE.DEPARTMENT_NUMBER
ORDER BY DEPTNO

--------
| TABLE SPACE SCAN - NO INDEX IS USED |
| STANDARD SEQUENTIAL PREFETCH WILL BE PERFORMED |
--------

KEY NO. COLUMN NAME COL.TYPE LNG NULL KEY CARD. ORDER LOW2KEY HIGH2KEY USED
--- ----------- -------- --- ---- ----- ----- ---------- ---------- ----
1 WORKDEPT CHAR 3 YES 8 ASC. C'B01 C'E11 <=**

--------

HOST VAR. TYPE LENGTH IND. HOST VARIABLE NAME
------------------------- --------- ----------------------------
 FIXED CHARACTER 4 NO HOSTVAR_STRUCTURE.DEPARTMENT_NUMBER
Accounting Report
Chapter 20. Bind Plan Data

This section of the Explain report is shown if the object being explained is a plan or query number for a mini plan created by the bind process. Information related to the binding of the plan, such as plan binder, bind time, or isolation level, is presented.

This part of the report is only produced in connection with the first SQL statement of the plan.

When a plan is explained using OMEGAMON XE for DB2 PE explain, the bind-related data is stored in a DB2 table which enables you to compare relevant information for up to three generations of the plan.

If, in a given row, there is a difference among the three columns, an arrow (<<) is shown in the rightmost column.

The bottom part of this section shows data for each DBRM and package in the most recent plan generation. The report states the precompilation date and time, programming language, number of SQL statements, single-byte or double-byte character set, use of comma, use of decimal(31), type of source, and the DB2 release when the module was precompiled.

For details on how to specify EXPLAIN commands refer to the Report Command Reference.

Explain Report - Plan Data Block

This is an example of the Bind Plan Data section.

<table>
<thead>
<tr>
<th>PLAN NAME: LOXXPLAN</th>
<th>LOCATION: SYSDSN5</th>
<th>CHANGES</th>
</tr>
</thead>
<tbody>
<tr>
<td>BOUND BY : IEN</td>
<td>QUALIFIER: PMDEV</td>
<td>BASE SIZE: 2040</td>
</tr>
<tr>
<td>AVG. SIZE: 0</td>
<td>CACHESIZE: 1024</td>
<td>PLENTRIES: 1</td>
</tr>
<tr>
<td>SQL STMTS: 7</td>
<td>SYS.ENTR. : 0</td>
<td>SQL STMTS: 7</td>
</tr>
<tr>
<td>VALIDATE : BIND</td>
<td>ISOLATION: CUR.STAB.</td>
<td></td>
</tr>
<tr>
<td>VALD : YES</td>
<td>OPERATIVE : YES</td>
<td></td>
</tr>
<tr>
<td>ACQUIRE : USE</td>
<td>RELEASE : COMMIT</td>
<td></td>
</tr>
<tr>
<td>DEFERPREP : NO</td>
<td>CURR. SERV: N/P</td>
<td></td>
</tr>
<tr>
<td>CURR. SERV: N/P</td>
<td>&lt;***</td>
<td></td>
</tr>
<tr>
<td>DEGREE : 1</td>
<td>REOPTIM. : NO</td>
<td></td>
</tr>
<tr>
<td>DYN. RULES: 0</td>
<td>KEEP DYN. : NO</td>
<td></td>
</tr>
<tr>
<td>SQLRULES : DB2</td>
<td>SDFF PROT. : DRA</td>
<td></td>
</tr>
<tr>
<td>PATH :</td>
<td>FNCT. RES. : 2002-07-15-12</td>
<td></td>
</tr>
<tr>
<td>DISCONNNET: EXPLICIT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OPTINTID: 0</td>
<td>STORED BY: PMDEV</td>
<td></td>
</tr>
<tr>
<td>STORED AT: 2002-01-15</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

-----------------------------------------------------------------------------------
Explain Report - Bind Plan Data
Chapter 21. Bind Package Data

This section of the Explain report is shown if the object being explained is a package.

The information related to the binding of the package, such as package owner, bind time or isolation level, is presented.

This part of the report is only produced in connection with the first SQL statement of the package.

For details on how to specify Explain commands refer to the Report Command Reference.

Explain Report - Package Data Block

This is an example of the Bind Package data section.

LOCATION : PMODA11
COLLECTION ID: KO2EX520
PACKAGE ID : DGO@TPG3
VERSION ID : OME_Final
CONSIST.TOKEN: X'18C8F8D602A50FF6'
POSSNAME : DB10MPE.BASE51.TKO2DBRM
OWNER : PMDEV52 QUOTE : APOSTROPHE
CREATOR : XRK COMMA : PERIOD
BIND DATE : 2013-02-15 HOSTLANG : ASSEMBLER
BIND TIME : 18.50.11.670393 CHARSET : ALPHANUMERIC
CREATE DATE : 2013-02-15 MIXED : NO
CREATE TIME : 18.49.53.347696 DEC31 : NO
QUALIFIER : PMDEV52 DATA CURRENCY: ALLOW BLOCKING
BASE SIZE : 5056 SQLERROR : NOPACKAGE
AVERAGE SIZE : 47760 SOURCE : DBRM
SYSENTRIES : 0 PRECOMP. DATE: 2010-07-15
SQL STATEMENT: 17 PRECOMP. TIME: 13.20.55.105599
VALIDATE : BIND VALID : YES
ISOLATION : UNCOMMITTED READ OPERATIVE : YES
RELEASE : CHECK PLAN REOPTIMIZAT. : NO
DEGREE : 1 DEFFPREPARE : INHERITED FROM PLAN
KEEP DYNAMIC : ORDA DDF PROTOCOL : INHERITED FROM PLAN
TYPE OF PACK.: BIND PACKAGE OPT_HINT_ID :
FNCT.RESOLVED: 2013-02-16-18.50.11.556346
...........5...........5...........5...........5...........5...........5...........5...........5...........5...........5
PATH: xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
Accounting Report
Chapter 22. Summary Report

This section introduces the summary report.

At normal completion of OMEGAMON XE for DB2 PE explain, the last pages of the output are a summary of the OMEGAMON XE for DB2 PE explain execution. This is called the Summary Report. Each command is prefixed with the report number.

For each SQLSTMT, QUERYNO, and QMFQUERY request, the access path is listed adjacent to the input request. For each PLAN or PACKAGE statement, the access path is listed for every SQL statement in the plan or package.

To the right of the access path information, the report page number (rrr-nnnn) is shown, unless LEVEL(SUMMARY) was specified. In this way the summary report can be used as a table of contents.

For details on how to specify EXPLAIN SUMMARY commands refer to the Report Command Reference.

Explain Report - Summary Report Block

This is an example of a Summary report.

REPORT ON: 02/18/13 15:05:08 OMEGAMON XE DB2 PE (V5.2) PAGE : SUMMARY
EXPLAIN SUMMARY REPORT USER AUTHID: XRK

THE FOLLOWING 1 EXPLAIN REQUESTS WERE PROCESSED: PAGE NO
1: DA11 PACKAGE : KO2EX520 .DG00TPG3
DETAIL REPORT REQUESTED
DBRM/PACK STMT TYP
DG00TPG3 1011020 P MATCHING INDEX SCAN(2/4)-DATA PAGES 1-2
DG00TPG3 1011021 P S TABLE SPACE SCAN-NO INDEX WILL BE USED 1-7
DG00TPG3 1011070 P L MATCHING INDEX SCAN(3/4)-DATA PAGES 1-12
DG00TPG3 1011071 P ADDITIONAL SORT FOR ORDER BY 1-16
DG00TPG3 1011071 P ADDITIONAL SORT FOR ORDER BY 1-20

Column description

The TYP column can contain the following values:

P Package
D DBRM
S Standard sequential prefetch
L Prefetch through a page list Table space scan
+ Nonmatching index scan

A blank No prefetch or others

The numbers in brackets following MATCHING INDEX SCAN, for example (2/3), show how many columns match, namely 2, and how many columns the index has, namely 3.
Chapter 23. EXPLAIN PLAN Command

This section introduces the EXPLAIN PLAN command.

When you bind a plan, you might decide to include a package list that contains wildcard characters, for example COLLID3.* or even *.*. This could result in a plan pointing at thousands of packages with an even greater number of explainable SQL statements. To control the volume of output produced, the PACKLIMIT option is provided.

If a particular plan consists of more packages than specified in PACKLIMIT, a report with all the package names is produced, but no SQL statements in these packages are explained. This report shows the collection IDs, the creator and owner names, the version IDs, and the dates of precompilation. However, if any DBRMs belong to the plan, SQL statements in these DBRMs are explained.

For details on how to specify EXPLAIN PLAN commands refer to the Report Command Reference.

Explain PLAN Report - Package List Block

The following report example results of an EXPLAIN PLAN statement with PACKLIMIT(10) if a plan named LARGPLAN was bound with a package list of COLLECT.*, which includes 80 packages:

```
ACTUAL AT: 01/30/13 13:10:38 OMEGAMON XE for DB2 PE (V5.2) PAGE : 1-50
EXPLAIN PLAN DB2 VERSION: V10
LOCATION : DSNAPC1 LARGEPLAN USER AUTHID: XRK
SUBSYSTEM: APC1 DETAIL CURR.SQLID : XRK

USE PACKAGE=COLLID.NAME.(VERSION) TO GET A DETAILED LISTING

<table>
<thead>
<tr>
<th>PACKAGE</th>
<th>COLLECTION ID.</th>
<th>CREATOR</th>
<th>OWNER</th>
<th>EXP</th>
<th>PC-DATE</th>
<th>VERSION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORG2UPT</td>
<td>XXASP16XXASP16</td>
<td>XXASP16</td>
<td>NO</td>
<td>1997-11-30 VERSION_1.0.0_27/1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ORG2UPT</td>
<td>XXASP16XXASP16</td>
<td>XXASP16</td>
<td>NO</td>
<td>1997-11-27 VERSION_1.0.0_20/1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ABINDCS</td>
<td>XXASP09XXASP09</td>
<td>XXASP09</td>
<td>NO</td>
<td>1997-07-24 NAMIK_PRIVATE_VER1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AIAPCS</td>
<td>XXASP09XXASP09</td>
<td>XXASP09</td>
<td>YES</td>
<td>1997-12-01 NAMIK_PRIVATE_VER1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RUCURH</td>
<td>XXASP09XXASP09</td>
<td>XXASP09</td>
<td>NO</td>
<td>1997-08-07 NAMIK_PRIVATE_VER1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHOLZ</td>
<td>USR1USR1</td>
<td>USR1</td>
<td>YES</td>
<td>2002-08-18 VER3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHOLZ</td>
<td>USR1USR1</td>
<td>USR1</td>
<td>YES</td>
<td>2002-08-18 VER2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHOLZ</td>
<td>USR1USR1</td>
<td>USR1</td>
<td>YES</td>
<td>2002-08-18 VER1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DB0C4</td>
<td>USR1USR1</td>
<td>USR1</td>
<td>YES</td>
<td>2002-08-18 VER3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DB0C4</td>
<td>USR1USR1</td>
<td>USR1</td>
<td>YES</td>
<td>2002-08-18 VER3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DB0C4</td>
<td>USR1USR1</td>
<td>USR1</td>
<td>YES</td>
<td>2002-08-18 VER3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DB0C4</td>
<td>USR1USR1</td>
<td>USR1</td>
<td>YES</td>
<td>2002-08-18 VER3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HVAR2</td>
<td>USR1USR1</td>
<td>USR1</td>
<td>YES</td>
<td>2002-08-18 VER3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HVAR2</td>
<td>USR1USR1</td>
<td>USR1</td>
<td>YES</td>
<td>2002-08-18 VER3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>KEYT</td>
<td>USR1USR1</td>
<td>USR1</td>
<td>YES</td>
<td>2002-08-18 VER3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>KEYT</td>
<td>USR1USR1</td>
<td>USR1</td>
<td>YES</td>
<td>2002-08-18 VER3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MIX</td>
<td>USR1USR1</td>
<td>USR1</td>
<td>YES</td>
<td>2002-08-18 VER3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MIX</td>
<td>USR1USR1</td>
<td>USR1</td>
<td>YES</td>
<td>2002-08-18 VER3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MIX</td>
<td>USR1USR1</td>
<td>USR1</td>
<td>YES</td>
<td>2002-08-18 VER3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MIX</td>
<td>USR1USR1</td>
<td>USR1</td>
<td>YES</td>
<td>2002-08-18 VER3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MX</td>
<td>USR1USR1</td>
<td>USR1</td>
<td>YES</td>
<td>2002-08-23 VER3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MX</td>
<td>USR1USR1</td>
<td>USR1</td>
<td>YES</td>
<td>2002-08-23 VER3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MX</td>
<td>USR1USR1</td>
<td>USR1</td>
<td>YES</td>
<td>2002-08-23 VER3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```
Accounting Report
Chapter 24. EXPLAIN PACKAGE Command

This section introduces the EXPLAIN PACKAGE command.

When a package is explained, you can specify the collection ID, the package name, or both, as generic names using an asterisk (*) as a wildcard character. For example, you have the following options:

```
EXPLAIN PACKAGE (DSNAPC1.COLLECT.MYT*)
EXPLAIN PACKAGE (DSNAPC1.COLLECT.MYTEST)
```

If a wildcard character is used, OMEGAMON XE for DB2 PE explain counts the total number of SQL statements that belong to the packages (the total number of SQL statements also includes nonexplainable statements).

If the number of packages is more than one, and the total number of SQL statements in these packages is larger than 300, OMEGAMON XE for DB2 PE explain does not explain any SQL statements unless the FORCE(YES) option is specified.

If the package specification does not contain any wildcard character, the total number of SQL statements is less than 300, or only one package conforms to the specification, the SQL statements are explained. However, if the package exists in more than one version, only the most recent version is explained. In this case a report showing all the versions for that package is produced.

For example, the following command was used to produce the report shown in "Explain Report - Package Version List Block" section:

```
EXPLAIN PACKAGE (DSNAPC1.COLLECT.MYTEST.(-7)) GEN(3)
```

For details on how to specify EXPLAIN commands refer to the Report Command Reference.

Explain Report - Package Version List Block

Here is an example of the Explain report for the Package Version List block:

```
FPEY0166I  PACKAGE MYTEST IN COLLECTION COLLECT HAS THE FOLLOWING VERSIONS

PRE-COMP'D EXP GEN  VERSION IDENTIFICATION
---------------------
2002-08-24 YES -01 VERSION1
2002-08-22 YES -02 VERSION2
2002-08-21 YES -03 VERSION3
2002-08-20 YES -04 VERSION4
2002-08-19 YES -05 VERSION5
2002-08-18 YES -06 VERSION6
2002-08-17 YES -07 VERSIONIDENT
2002-08-16 YES -08 VERSION8
2002-08-15 YES -09 VERSION9
2002-08-14 YES -10 < VERSION IDENTIFICATION NOT SPECIFIED >
2002-08-13 YES -11 VERSION11
2002-08-12 YES -12 VERSION12
2002-08-11 YES -13 VERSION13
2002-08-10 YES -14 VERSION14
2002-08-09 YES -15 VERSION15
2002-08-08 YES -16 VERSION16
2002-08-07 YES -17 VERSION17
```

© Copyright IBM Corp. 1985, 2013
This report shows that the package exists in 21 versions, where the version that corresponds to generation number -7 was selected for explanation. The report further shows that the user asked for the explanation of three generations (-7, -8, and -9). The first generation to be explained is marked with an asterisk (*). Instead of specifying a generation ID, the version ID could be specified. The version ID can be specified either in full or in combination with a wildcard character as in the following example:

```
EXPLAIN PACKAGE(OSNAPC1.COLLECT.MYTEST.(VERSION7*)) GEN(1)
```

A maximum of 100 generations are listed.
Chapter 25. EXPLAIN SQLSTMT Command

This section shows an example of an Explain report generated with the EXPLAIN SQLSTMT command.

For details on how to specify EXPLAIN commands refer to the Report Command Reference.

Explain Report - Example for SQL Text

Use the following EXPLAIN SQLSTMT command to generate the following example of an explain report.

```
GLOBAL PLANEX(KO2EXPL)
SSID(DA11)
SQLID(XRK)
EXPLAIN SQLSTMT
| SELECT * FROM SYSIBM.SYSPACKAGE
  WHERE NAME = 'DGO@TPG3'
  AND COLLID = 'KO2EX520' |
LEVEL(INDEXES)
EXEC
```

Here is an example of an SQL Text.

ACTUAL AT: 02/21/13 11:57:21 OMEGAMON XE DB2 PE (V5.2) PAGE : 1-1
LOCATION : PMODA11 USER AUTHID: XRK
SUBSYSTEM: DA11 INDEX CURR.SQLID : XRK

SQL STATEMENT TEXT :
```
SELECT * FROM SYSIBM.SYSPACKAGE
WHERE NAME = 'DGO@TPG3'
AND COLLID = 'KO2EX520'
```

-----------------------------------------------------------------------------
Explain Table: XRK.DSN_STATEMNT_TABLE -----------------------------------------
Explain Time: 2013-02-21-11.57.21.680000
Program : GD0RTPT2 , Collid : KO2EX520
Version : OMPX_FINAL , Apiname (Plan) : N/P
Queryno : 999735911 , Sectnoi : 0
Stm_encode : U - Unicode , Stm_type : SELECT
Procs (Cost MS) : 1 , Cost_category : N/P
Procsu (Cost SU) : 11 , Reason (Category) : N/P
Total_cost : 1 , Group_member : N/P
Cost_category : A - Cost estimate without using default values
Explain Table: XRK.PLAN_TABLE -------------------------------------------------
Explain Time: 2013-02-21-11.57.21.680000
Timestamp : 2013-02-21-11:57:21.68
Program : GD0RTPT2 , Collid : KO2EX520
Version : OMPX_FINAL , Apiname (Plan) : N/P
Queryno : 999735911 , Sectnoi : 0
Plan_no : 1 , Parent_plan_no : 0
Mixopseq : 0 , Qblock_type : SELECT
Tname (Table) : SYSIBM.SYSPACKAGE , Creator (Table) : SYSIBM
Table (Table) : 1 , Correlation_name : N/P
Table_type : T - Table , Tcref : 0
Table_encode : U - Unicode , Table_mccsid : 1200
Table_sccsid : 367 , Table_DCCSID : 1200
Tslmode : N - No lock * , Group_member : N/P
Access_type : I - Index scan , Primary_access_type : BLANK
Accessname (Index) : DSNKX01 , Accesscreator : SYSIBM
Matchcols : 0 , Indexonly : N/O
Method (Join) : 0 - First table , Join_degree : 0
Join_type : b - INNER or NO , Merge : N/O
Merge_join_cols : 0 , Merge : N/O
Prefetch : S - Sequential , Page_range : N/O
WHEN_OPTIMIZE : b - At bind time, ACCESS_DEGREE : 0
COLUMNFN_EVAL : BLANK, ROUTINE_ID : 0
HINT_USED : N/P, OPTHINT : N/P
SCAN_DIRECTION : N/A
SORTN_PGROUP_ID : 0, SORTN_UNIQUE : NO, SORTC_UNIQUE : NO
SORTC_PGROUP_ID : 0, SORTN_JOIN : NO, SORTC_JOIN : NO
ACCESS_PGROUP_ID : 0, SORTN_ORDERBY : NO, SORTC_ORDERBY : NO
JOIN_PGROUP_ID : 0, SORTN_GROUPBY : NO, SORTC_GROUPBY : NO
REMARKS : N/P, STMTTOKEN : N/P
PARALLELISM_MODE : BLANK, BIND EXPLAIN ONLY : NO
EXPANSION_REASON : N/A
TSLOCKMODE : N - No lock (UR isolation)

THE ACCESS PATH CHOSEN BY DB2 AT 11:57:21.6 ON 2013-02-21
+------------------------------------------------------------------+
| NON-MATCHING INDEX SCAN WITH SCAN OF REFERENCED DATA PAGES       |
| PURE SEQUENTIAL PREFETCH WILL BE PERFORMED                      |
| PAGE RANGE SCAN WILL NOT BE USED                               |
+------------------------------------------------------------------+
INDEX: SYSIBM.DSNKKX01---------------------------------------------
STATSTIME: 2013-02-16-14.45.54.186869
CREATED : 0001-01-01-00.00.00.000000 ALTERED: 2003-09-21-23.30.17.962937
FULL KEY CARD: 1,SPACE : 245.760K,UNIQUE: YES,CLUSTERED: N
INDEX TYPE : 2,MSGIZE: 4096,BPOOL: BP0,DB.NAME : DSNDB06
CLUSTERRATIO : 84.6328%,ERRULE: NO,CLRULE: NO,IXSPACE : DSNKKX01
MAX.PIECESIZE: 0,COPY : NO,COPYLRSN: X'000000000000'

TABLE: SYSIBM.SYSPACKAGE -----------------------------------------
STATSTIME: 2013-02-16-14.45.54.186869, TB TYPE : TABLE
CREATED : 1985-04-01-00.00.00.000000, ALTERED : 2010-07-01-09.00.57.417442
ROWS : 885, COLUMNS : 60, ROWLENGTH: 3913, EDIT PROC.: %
TAB.STAT.: ENC.SCHEME: UNICODE
TABLESPACE: DSNDB06.SYSTSPKG --------------------------------------
NAME : SYSTSPKG , DATABASE : DSNDB06
CREATOR : SYSIBM , CREATED BY: SYSIBM
STATSTIME : 2013-02-16-14.45.54.186869,
ACTIVE PGS: 372, DBASE ID : 6, TS STATUS : A, TS TYPE : G
PAGE SIZE : 4096, OBJ ID : 207, ERASURE: NO, STORGROUP : SPACE : 1.720M, PAGESET ID: 2068, CLOSERULE : NO, BUFPOOL : BP0
DB SIZE : 68.719G, OLD VERS : 0, LOCKPART : N/A, ENC.SCHEME: UNICODE
LOCKMAX : SYSTEM, CUR VERS : 0, SPACE : DP, DBCS CCSID: 1000
TABLES/TS : 1, PARTITIONS: 1, LOG : YES, DBCS CCSID: 1200
MAXROWS : 255, SEG SIZE : 32, IMPLICIT : NO, MIX. CCSID: 1200
AVG ROWLEN: 201,
REPORT ON: 02/21/13 11:57:21 OMEGAMON XE DB2 PE (V5.2) PAGE : SUMMARY
EXPLAIN SUMMARY REPORT USER AUTHID: XKR

THE FOLLOWING 1 EXPLAIN REQUESTS WERE PROCESSED: PAGE NO:

1 : DA11 SQL STMT
INDEXES REPORT REQUESTED
SELECT * FROM SYSIBM.SYSPACKAGE
WHERE NAME = 'DA11TPG3'
AND COLID = 'K02X520'
+ NON-MATCHING INDEX SCAN-DATA PAGES SCAN

OMEGAMON XE for DB2 PE (V5.2) EXPLAIN PROCESSING COMPLETED.
Part 5. The I/O Activity Report Set

These topics provide information about the I/O activity reports.

**Note:** For an introduction to the I/O Activity report set and general I/O Activity information refer to the [Reporting User’s Guide](#).

Chapter 26, “Summary and Detail Report Header,” on page 26-1

OMEGAMON XE for DB2 PE header information is printed at the top of each page of the summary and detail report.

Chapter 27, “I/O Activity Summary Report and Field Descriptions,” on page 27-1

The I/O activity summary report provides an overview of system-wide I/O activity that is used to monitor trends and identify problem areas.

Chapter 28, “I/O Activity Detail Reports,” on page 28-1

A separate detail report is produced for each category of I/O activity.
Chapter 26. Summary and Detail Report Header

OMEGAMON XE for DB2 PE header information is printed at the top of each page of the summary and detail report.

I/O Activity Report Header Example

Here is an example of an I/O Activity Report Header.

LOCATION: LOCATION1
GROUP: GROUP2
MEMBER: MEMBER4
SUBSYSTEM: DB2D
DB2 VERSION: V10

Field description

The report header contains the following information, described in the order left block, middle block, right block:

LOCATION
The DB2 reporting location. If the location name is not available, the DB2 data sharing group name is printed in this field. If the DB2 data sharing group name does not exist, the DB2 subsystem ID is printed.

GROUP
The name of the DB2 data sharing group. This field shows N/A if there is no group name.

MEMBER
The name of the DB2 data sharing member or the member name of the DB2 subsystem. This field shows N/A if there is no member name.

This field is not printed on the report page showing the GROUP BUFFERPOOLS PARAMETERS.

SUBSYSTEM
The ID of the DB2 subsystem that generated the data. This field is not printed on the report page showing the GROUP BUFFERPOOLS PARAMETERS.

DB2 VERSION
The DB2 version number of the subsystem that generated the data.

OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (VnRnMn)
The product name and the version, release, and modification level.

Title - layout
The title of the report and the layout. The layout can be a default layout provided with OMEGAMON XE for DB2 PE or a layout you have tailored yourself.

ORDER
If the ORDER option of the REPORT or TRACE subcommand was used to arrange the report entries, the selected keywords are shown in this field. Depending on the context, the OMEGAMON XE for DB2 PE identifiers by which lock events are grouped are shown here.

PAGE
The page number in the format lll-nnnnnn, where lll denotes the location number within the report and nnnnnn the page number within the location.
REQUESTED FROM and TO
The FROM and TO dates and times specified in the REPORT or TRACE subcommand.

If both FROM and TO dates and times are omitted from the REPORT subcommand, the FROM and TO dates and times specified in GLOBAL are printed. If only the FROM date and time or only the TO date and time has been specified, NOT SPECIFIED is printed for the unspecified value.

If FROM and TO are not specified in REPORT or GLOBAL, NOT SPECIFIED appears for both the FROM and TO values.

If you have specified FROM and TO times without dates in REPORT or GLOBAL, ALL DATES is printed along with the specified times.

INTERVAL FROM
The start date and time of the first reduction interval covered by the report. If REDUCE is not specified, the INTERVAL defaults to 0 and the timestamps of the first and last events are printed.

INTERVAL TO
The end date and time of the last reduction interval covered by the report. If REDUCE is not specified, the INTERVAL defaults to 0 and the timestamps of the first and last events are printed.
Chapter 27. I/O Activity Summary Report and Field Descriptions

The I/O activity summary report provides an overview of system-wide I/O activity that is used to monitor trends and identify problem areas.

The I/O activity summary report:
- Summarizes the information contained in the I/O activity detail reports for a location (refer to Chapter 28, “I/O Activity Detail Reports,” on page 28-1).
- Shows on a single page a block of entries for each of the I/O categories: buffer pool, EDM pool, active log, archive log/BSDS, and cross invalidation (XI).
- Is produced if you specify the IOACTIVITY(REPORT) command and there is at least one I/O activity IFCID begin/end pair in the input data set satisfying the FROM and TO, and INCLUDE or EXCLUDE criteria. You do not have to specify the SUMMARY level because this is the default.

The following command produces the I/O activity summary report shown in “Example I/O Activity Summary Report.”

```
IOACTIVITY REPORT
```

Example I/O Activity Summary Report

Here is an example of an I/O Activity Summary report.

```
LOCATION: DSNCAT OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 2-1
GROUP: DSNCAT I/O ACTIVITY REPORT - SUMMARY REQUESTED FROM: NOT SPECIFIED
MEMBER: V71B TO: NOT SPECIFIED
SUBSYSTEM: V71B INTERVAL FROM: 01/30/10 19:32:54.82 DB2 VERSION: V10 TO: 01/30/10 19:56:28.53

CT/PT/DBD NOT IN AVG LEN

BUFFER POOL TOTALS AET EDM POOL REFERENCES EDM POOL AET (BYTES)

TOTAL I/O REQUESTS 51 0.019885 CURSOR TABLE - HEADER 1 2 0.131381 7656.00
CURSOR TABLE - DIRECTORY 1 1 0.000568 228.00
TOTAL READ I/O REQUESTS 51 0.019885 CURSOR TABLE - RDS SECTION 9 9 0.022214 1475.11
NON-PREFETCH READS 51 -- TOTAL PLANS ------ 11 12 0.030170 1923.64
PREFETCH READS
WITHOUT I/O 0 PACKAGE TABLE - HEADER 0 0 N/C 0.00
WITH I/O 0 PACKAGE TABLE - DIRECTORY 0 0 N/C 0.00
PAGES READ 0 PACKAGE TABLE - RDS SECTION 0 0 N/C 0.00
PAGES READ / SUCC READ 0.00 -- TOTAL PACKAGES -- 0 0 N/C 0.00
TOTAL WRITE REQUESTS 0 N/C DATABASE DESCRIPTORS 0 0 N/C 0.00
SYNCHRONOUS WRITES 0 N/C
COUPLING FACILITY CASTOUTS 0 N/C
PAGES WRITTEN PER WRITE 0.00
ASYNCHRONOUS WRITES 0 N/C
COUPLING FACILITY CASTOUTS 0 N/C
PAGES WRITTEN PER WRITE 0.00

ACTIVE LOG TOTALS AET ARCHIVE LOG TOTALS AET

TOTAL WAITS 22 0.018384 READ REQUESTS 0 N/C READ REQUESTS 0 N/C 13 0.078868
READ REQUESTS 0 N/C DASD READ 0 N/C
WRITE REQUESTS 22 0.018384 OFFLOAD REQUESTS 0 N/C WRITE REQUESTS 12 0.022673
CONT. CI / WRITE 1.00 TAPE READ 0 N/C
OTHER WAITS 0 N/C BLOCKS / OFFLOAD 0.00
ALLOCATE 0 N/C CROSS-INVALIDATION ACTIVITY TOTALS

© Copyright IBM Corp. 1985, 2013
## I/O Activity - Summary Report

<table>
<thead>
<tr>
<th>Action</th>
<th>Count</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEALLOCATE</td>
<td>0</td>
<td>N/C</td>
</tr>
<tr>
<td>OPEN</td>
<td>0</td>
<td>N/C</td>
</tr>
<tr>
<td>CLOSE</td>
<td>0</td>
<td>N/C</td>
</tr>
<tr>
<td>HSM RECALL</td>
<td>0</td>
<td>N/C REFRESHED FROM DASD 0</td>
</tr>
<tr>
<td>CATALOG LOCATE</td>
<td>0</td>
<td>N/C REFRESHED FROM DASD 0</td>
</tr>
<tr>
<td>MULTI-DATA SET TAPE</td>
<td>0</td>
<td>N/C REFRESHED FROM GROUP BPOOL 0</td>
</tr>
<tr>
<td>TAPE VOL POSITIONING</td>
<td>0</td>
<td>N/C REFRESHED FROM GROUP BPOOL 0</td>
</tr>
<tr>
<td>WTOR ISSUED</td>
<td>0</td>
<td>N/C REFRESHED FROM DASD 0</td>
</tr>
<tr>
<td>DATA SET UNAVAILABLE</td>
<td>0</td>
<td>N/C</td>
</tr>
<tr>
<td>PHYSICAL UNIT UNAV.</td>
<td>0</td>
<td>N/C</td>
</tr>
<tr>
<td>RDR SERV.UNAVAILABLE</td>
<td>0</td>
<td>N/C</td>
</tr>
</tbody>
</table>

---

I/O ACTIVITY REPORT COMPLETE

**“BUFFER POOL” on page 27-3**
This section of the summary report contains selected fields from the Buffer Pool report.

**“EDM POOL” on page 27-5**
This section of the summary report contains selected fields from the EDM pool report.

**“ACTIVE LOG” on page 27-7**
This section of the summary report contains selected fields from the Active Log report.

**“ARCHIVE LOG” on page 27-8**
This section of the summary report contains selected fields from the Archive Log report.

**“BOOTSTRAP DATASET” on page 27-10**
This section of the summary report contains selected fields from the Bootstrap Dataset report.

**“CROSS-INVALIDATION ACTIVITY” on page 27-11**
This section of the summary report contains selected fields from the Cross-Invalidation report.
This section of the summary report contains selected fields from the Buffer Pool report.

The following fields are printed in the report:

**TOTAL I/O REQUESTS**

The total number of I/O requests (TOTALS) and the average elapsed time per I/O request (AET).

**TOTAL READ I/O REQUESTS**

The total number of I/O read requests (TOTALS) and the average elapsed time per read request (AET).

**NON-PREFETCH READS**

The total number of non-prefetch reads.

**PREFETCH READS**

An aggregate of all types of prefetches:
- Sequential prefetches (determined at bind time)
- List prefetch
- Sequential prefetch triggered by the sequential detection logic

**WITHOUT I/O**

The number of unsuccessful prefetch reads. This can occur because all the pages requested by a prefetch read were already in the buffer pool.

**WITH I/O**

The number of successful prefetch reads.

**PAGES READ**

The number of pages read for all prefetch read requests.

**PAGES READ / SUCC READ**

The number of pages read per successful prefetch read request.

**TOTAL WRITE REQUESTS**

The total number of write I/O requests (TOTALS) and the average elapsed time per write request (AET).

The write requests are divided into two categories: synchronous writes and asynchronous writes.

**SYNCHRONOUS WRITES**

The number of synchronous writes (TOTALS) and the average elapsed time per synchronous write request (AET).

**COUPLING FACILITY CASTOUTS**

The number of synchronous writes due to coupling facility castouts.

**PAGES WRITTEN PER WRITE**

The average number of pages per synchronous write.

**ASYNCHRONOUS WRITES**

The number of asynchronous writes (TOTALS) and the average elapsed time per asynchronous write (AET).
I/O Activity - Summary Report

COUPLING FACILITY CASTOUTS
The number of asynchronous writes due to coupling facility castouts.

PAGES WRITTEN PER WRITE
The average number of pages written per asynchronous write.
EDM POOL

This section of the summary report contains selected fields from the EDM pool report.

The following fields are printed in the report:

**CURSOR TABLE - HEADER**
- The number of load requests for cursor table headers (CT/PT/DBD REFERENCES).
- The number of cursor table header loads not in the EDM pool (NOT IN EDM POOL).
- The average elapsed time of a cursor table header load (AET).
- The average length of a loaded cursor table header in bytes (AVG LEN (BYTES)).

**CURSOR TABLE - DIRECTORY**
- The number of load requests for cursor table directories (CT/PT/DBD REFERENCES).
- The number of cursor table directory loads not in the EDM pool (NOT IN EDM POOL).
- The average elapsed time of a cursor table directory load (AET).
- The average length of a loaded cursor table directory in bytes (AVG LEN (BYTES)).

**CURSOR TABLE - RDS SECTION**
- The number of load requests for cursor table RDS sections (CT/PT/DBD REFERENCES).
- The number of cursor table RDS section loads not in the EDM pool (NOT IN EDM POOL).
- The average elapsed time of a cursor table RDS section load (AET).
- The average section length of a loaded cursor table RDS section in bytes (AVG LEN (BYTES)).

**TOTAL PLANS**
- The number of load requests for plans; that is, the sum of CT/PT/DBD references for cursor table header, directory, and RDS section (CT/PT/DBD REFERENCES).
- The number of plan loads not in the EDM pool (NOT IN EDM POOL).
- The average elapsed time per plan load (AET).
- The average section length of a loaded plan in bytes (AVG LEN (BYTES)).

**PACKAGE TABLE - HEADER**
- The number of load requests for package table headers (CT/PT/DBD REFERENCES).
- The number of package table header loads not in the EDM pool (NOT IN EDM POOL).
- The average elapsed time of a package table header load (AET).
- The average length of a loaded package table header in bytes (AVG LEN (BYTES)).
I/O Activity - Summary Report

PACKAGE TABLE - DIRECTORY
The number of load requests for package table directories (CT/PT/DBD REFERENCES).
The number of package table directory loads not in the EDM pool (NOT IN EDM POOL).
The average elapsed time of a package table directory load (AET).
The average length of a loaded package table directory in bytes (AVG LEN (BYTES)).

PACKAGE TABLE - RDS SECTION
The number of load requests for package table RDS sections (CT/PT/DBD REFERENCES).
The number of package table RDS section loads not in the EDM pool (NOT IN EDM POOL).
The average elapsed time of a package table RDS section load (AET).
The average section length of a package table RDS section in bytes (AVG LEN (BYTES)).

TOTAL PACKAGES
The number of load requests for package tables; that is, the sum of CT/PT/DBD references for the package table header, directory, and RDS section (CT/PT/DBD REFERENCES).
The number of package table loads not in the EDM pool (NOT IN EDM POOL).
The average elapsed time per package table load (AET).
The average section length of a loaded package table in bytes (AVG LEN (BYTES)).

DATABASE DESCRIPTORS
The number of load requests for database descriptors (CT/PT/DBD REFERENCES).
The number of DBD loads not in the EDM pool (NOT IN EDM POOL).
The average elapsed time of a database descriptor load (AET).
The average length of a loaded database descriptor in bytes (AVG LEN (BYTES)).
This section of the summary report contains selected fields from the Active Log report.

The following fields are printed in the report:

**TOTAL WAITS**
   The total number of waits for read and write requests and other waits (TOTALS) and the average elapsed time of an I/O wait of any type (AET).

**READ REQUESTS**
   The number of read requests (TOTALS) and the average elapsed time of a read request (AET).

**WRITE REQUESTS**
   The number of write requests (TOTALS) and the average elapsed time of a write request (AET).

**CONT. CI / WRITE**
   The number of contiguous control intervals per write request.

**OTHER WAITS**
   The total number of waits for resource allocation and deallocation, and the number of waits to open and close data sets (TOTALS).
   The average elapsed time of all other waits (AET).
   The next four rows relate to waits other than waits for read or write requests.

**ALLOCATE**
   The number of waits for resource allocation (TOTALS) and the average elapsed time of a wait for resource allocation (AET).

**DEALLOCATE**
   The number of waits for resource deallocation (TOTALS) and the average elapsed time of waits for resource deallocation (AET).

**OPEN**
   The number of waits to open data sets (TOTALS) and the average elapsed time of waits to open data sets (AET).

**CLOSE**
   The number of waits to close data sets (TOTALS) and the average elapsed time of waits to close data sets (AET).
ARCHIVE LOG

This section of the summary report contains selected fields from the Archive Log report.

The following fields are printed in the report:

READ REQUESTS
The total number of archive read requests (TOTALS) and the average elapsed time of an archive read request (AET).

Archive read requests are subdivided into the following categories:

DASD READ
The total number of reads from the DASD and (TOTALS) the average elapsed time of reads from the DASD (AET).

TAPE READ
The total number of reads from the tape (TOTALS) and the average elapsed time of reads from the tape (AET).

OFFLOAD REQUESTS
The total number of archive offloads (TOTALS) and the average elapsed time per archive offload (AET).

BLOCKS / OFFLOAD
The number of blocks of data written per offload.

OTHER WAITS
Waits other than read or write requests. The total elapsed time (TOTALS) and the average elapsed time of all other waits (AET).

ALLOCATE
The total number of waits (TOTALS) and the average elapsed time of a wait for resource allocation (AET).

DEALLOCATE
The total number of waits (TOTALS) and the average elapsed time of a wait for resource deallocation (AET).

OPEN
The total number of waits (TOTALS) and the average elapsed time of a wait to open data sets (AET).

CLOSE
The total number of waits (TOTALS) and the average elapsed time of a wait to close data sets (AET).

HSM RECALL
The total number of waits (TOTALS) and the average elapsed time of a wait for HSM to recall data sets (AET).

CATALOG LOCATE
The total number of waits (TOTALS) and the average elapsed time per wait to locate data sets through the catalog (AET).

MULTI-DATA SET TAPE
The total number of waits (TOTALS) and the average elapsed time per wait for multi-data set tape volume (AET).

TAPE VOL POSITIONING
The total number of waits (TOTALS) and the average elapsed time per wait for tape volume positioning (AET).
I/O Activity - Summary Report

**WTOR ISSUED**
The total number of waits (TOTALS) and the average elapsed time per wait due to write-to-operator messages being issued (AET).

**DATA SET UNAVAILABLE**
The total number of waits (TOTALS) and the average elapsed time of waits due to a data set being unavailable (AET).

**PHYSICAL UNIT UNAV.**
The total number of waits (TOTALS) and the average elapsed time of waits due to an unavailable physical unit (AET).

**RDR SERV. UNAVAILABLE**
The total number of waits (TOTALS) and the average elapsed time of a wait due to an unavailable reader service task (AET).
BOOTSTRAP DATASET

This section of the summary report contains selected fields from the Bootstrap Dataset report.

The following fields are printed in the report:

TOTAL WAITS
The total number of waits due to read and write requests for the bootstrap data set (TOTALS) and the average duration of bootstrap data set waits (AET).

READ REQUESTS
The total number of BSDS reads (TOTALS) and the average elapsed time per read from BSDS (AET).

WRITE REQUESTS
The total number of writes to the BSDS (TOTALS) and the average elapsed time per BSDS write (AET).
CROSS-INVALIDATION ACTIVITY

This section of the summary report contains selected fields from the Cross-Invalidation report.

The following fields are printed in the report:

SYNCHRONOUS READS
  The number of cross-invalidated pages which are refreshed via synchronous read.
  
  REFRESHED FROM GROUP BPOOL
  The number of cross-invalidated pages which are refreshed from the group buffer pool via synchronous read.

  REFRESHED FROM DASD
  The number of cross-invalidated pages which are refreshed from the DASD via synchronous read.

SEQUENTIAL PREFETCH
  The number of cross-invalidated pages which are refreshed via sequential prefetch.

  REFRESHED FROM GROUP BPOOL
  The number of cross-invalidated pages which are refreshed from the group buffer pool via sequential prefetch.

  REFRESHED FROM DASD
  The number of cross-invalidated pages which are refreshed from the DASD via sequential prefetch.
Accounting Report
Chapter 28. I/O Activity Detail Reports

A separate detail report is produced for each category of I/O activity.

Totals are accumulated for most columns in the report. The information in the Total column on each detail report for grand totals also appears in the I/O activity summary report.

The following sections show samples and field descriptions for each category of the I/O Activity Detail report.

"Buffer Pool Report" on page 28-2
"EDM Pool Report" on page 28-6
The EDM pool report provides information about the number of cursor table, package table, or database directory requests, loads from the DASD, their average elapsed times, and the average section lengths of the loaded data items.

"Active Log Report" on page 28-9
The active log report provides information about the writing and retrieving of log records. It presents the number of reads, writes, and non-I/O waits related to the active log and the average elapsed times spent waiting for these events.

"Archive Log/BSDS Report" on page 28-12
The archive log/BSDS report provides information about the writing of log records and the retrieval of log data. It also contains information about the bootstrap data set that controls the movement of full active log data sets to the archive log.

"Cross-Invalidation Report" on page 28-19
The cross-invalidation report presents buffer refresh events due to cross invalidation summarized by selected OMEGAMON XE for DB2 PE identifiers. If two DB2 systems compete for read/write interest on a page set or partition, a certain amount of buffer cross-invalidation activity occurs to maintain DB2 buffer pool coherency between the two systems.
The buffer pool report provides information about the movement of database pages between the DASD and the main storage buffers. It presents information about the number of read and write operations, the amount of read and write requests, and the average wait times.

The following command produces the buffer pool report shown in “I/O Activity Buffer Pool Report.”

The following example shows a buffer pool report.

### I/O Activity Buffer Pool Report

The following is a detailed report of I/O activity for a specific buffer pool.

<table>
<thead>
<tr>
<th>INTERVAL</th>
<th>PRIMAUTH</th>
<th>PLANNAME</th>
<th>TOTAL AET</th>
<th>TOTAL TYPE AET</th>
<th>% READ I/O AET</th>
<th>TOTAL TYPE OUT AET</th>
<th>WRITE OPEN</th>
</tr>
</thead>
<tbody>
<tr>
<td>01/30 00:00 - 06/01 00:05</td>
<td>AUTH_10</td>
<td>PLAN_10</td>
<td>8</td>
<td>0.028750</td>
<td>1 SYNCH</td>
<td>0.00</td>
<td>100.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.00</td>
<td>0.00</td>
<td>100.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.00</td>
<td>0.00</td>
<td>100.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.00</td>
<td>0.00</td>
<td>100.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.00</td>
<td>0.00</td>
<td>100.00</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>INTERVAL</th>
<th>PRIMAUTH</th>
<th>PLANNAME</th>
<th>TOTAL AET</th>
<th>TOTAL TYPE AET</th>
<th>% READ I/O AET</th>
<th>TOTAL TYPE OUT AET</th>
<th>WRITE OPEN</th>
</tr>
</thead>
<tbody>
<tr>
<td>01/30 00:05 - 06/01 00:10</td>
<td>AUTH_10</td>
<td>PLAN_10</td>
<td>4</td>
<td>0.085000</td>
<td>1 SYNCH</td>
<td>0.100000</td>
<td>100.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.100000</td>
<td>0.00</td>
<td>100.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.100000</td>
<td>0.00</td>
<td>100.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.100000</td>
<td>0.00</td>
<td>100.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.100000</td>
<td>0.00</td>
<td>100.00</td>
</tr>
</tbody>
</table>

---

### Grand Total

<table>
<thead>
<tr>
<th>INTERVAL</th>
<th>PRIMAUTH</th>
<th>PLANNAME</th>
<th>TOTAL AET</th>
<th>TOTAL TYPE AET</th>
<th>% READ I/O AET</th>
<th>TOTAL TYPE OUT AET</th>
<th>WRITE OPEN</th>
</tr>
</thead>
<tbody>
<tr>
<td>01/30 00:00 - 06/01 00:05</td>
<td>AUTH_10</td>
<td>PLAN_10</td>
<td>12</td>
<td>0.047500</td>
<td>2 SYNCH</td>
<td>0.100000</td>
<td>50.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.100000</td>
<td>0.00</td>
<td>100.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.100000</td>
<td>0.00</td>
<td>100.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.100000</td>
<td>0.00</td>
<td>100.00</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>INTERVAL</th>
<th>PRIMAUTH</th>
<th>PLANNAME</th>
<th>TOTAL AET</th>
<th>TOTAL TYPE AET</th>
<th>% READ I/O AET</th>
<th>TOTAL TYPE OUT AET</th>
<th>WRITE OPEN</th>
</tr>
</thead>
<tbody>
<tr>
<td>01/30 00:05 - 06/01 00:10</td>
<td>AUTH_10</td>
<td>PLAN_10</td>
<td>4</td>
<td>0.085000</td>
<td>2 SYNCH</td>
<td>0.100000</td>
<td>100.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.100000</td>
<td>0.00</td>
<td>100.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.100000</td>
<td>0.00</td>
<td>100.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.100000</td>
<td>0.00</td>
<td>100.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.100000</td>
<td>0.00</td>
<td>100.00</td>
</tr>
</tbody>
</table>

---

### Grand Total

<table>
<thead>
<tr>
<th>INTERVAL</th>
<th>PRIMAUTH</th>
<th>PLANNAME</th>
<th>TOTAL AET</th>
<th>TOTAL TYPE AET</th>
<th>% READ I/O AET</th>
<th>TOTAL TYPE OUT AET</th>
<th>WRITE OPEN</th>
</tr>
</thead>
<tbody>
<tr>
<td>01/30 00:00 - 06/02 00:05</td>
<td>AUTH_10</td>
<td>PLAN_10</td>
<td>8</td>
<td>0.028750</td>
<td>1 SYNCH</td>
<td>0.00</td>
<td>100.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.00</td>
<td>0.00</td>
<td>100.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.00</td>
<td>0.00</td>
<td>100.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.00</td>
<td>0.00</td>
<td>100.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.00</td>
<td>0.00</td>
<td>100.00</td>
</tr>
</tbody>
</table>

---

The buffer pool report provides information about the movement of database pages between the DASD and the main storage buffers. It presents information about the number of read and write operations, the amount of read and write requests, and the average wait times.
**I/O Activity - Detail Report**

<table>
<thead>
<tr>
<th></th>
<th>PAGE</th>
<th>ACTIVE</th>
<th>UPDATED</th>
<th>FAULTS</th>
<th>BUFFERS</th>
<th>PAGES</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>PLAN_10</strong></td>
<td>8</td>
<td>0.02875</td>
<td>1.00</td>
<td>0.00</td>
<td>0.00</td>
<td>100.00</td>
</tr>
<tr>
<td></td>
<td>1 SYNCH</td>
<td>N/C</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>100.00</td>
</tr>
<tr>
<td></td>
<td>1 SEQPF</td>
<td>N/C</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>100.00</td>
</tr>
<tr>
<td></td>
<td>1 DYNPF</td>
<td>N/C</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>100.00</td>
</tr>
<tr>
<td></td>
<td>1 LSTPF</td>
<td>N/C</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>100.00</td>
</tr>
</tbody>
</table>

**PAGE ACTIVE UPDATED**

<table>
<thead>
<tr>
<th></th>
<th>FAULTS</th>
<th>BUFFERS</th>
<th>PAGES</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>7</td>
<td>167</td>
<td></td>
</tr>
</tbody>
</table>

**01/30 00:05 - 01/30 00:10**

<table>
<thead>
<tr>
<th></th>
<th>PAGE</th>
<th>ACTIVE</th>
<th>UPDATED</th>
<th>FAULTS</th>
<th>BUFFERS</th>
<th>PAGES</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>PLAN_10</strong></td>
<td>6</td>
<td>0.08500</td>
<td>1.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td>1 SYNCH</td>
<td>0.10000</td>
<td>100.00</td>
<td>1.00</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1 SEQPF</td>
<td>0.07000</td>
<td>100.00</td>
<td>10.00</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1 DYNPF</td>
<td>0.09000</td>
<td>100.00</td>
<td>30.00</td>
<td>0.00</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1 LSTPF</td>
<td>0.08000</td>
<td>100.00</td>
<td>20.00</td>
<td>0.00</td>
<td>0</td>
</tr>
</tbody>
</table>

***GRAND TOTAL***

<table>
<thead>
<tr>
<th></th>
<th>PAGE</th>
<th>ACTIVE</th>
<th>UPDATED</th>
<th>FAULTS</th>
<th>BUFFERS</th>
<th>PAGES</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>PLAN_10</strong></td>
<td>12</td>
<td>0.04750</td>
<td>1.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td>2 SYNCH</td>
<td>0.10000</td>
<td>50.00</td>
<td>1.00</td>
<td>50.00</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>2 SEQPF</td>
<td>0.07000</td>
<td>50.00</td>
<td>10.00</td>
<td>50.00</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>2 DYNPF</td>
<td>0.09000</td>
<td>50.00</td>
<td>30.00</td>
<td>50.00</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>2 LSTPF</td>
<td>0.08000</td>
<td>50.00</td>
<td>20.00</td>
<td>50.00</td>
<td>2</td>
</tr>
</tbody>
</table>

**I/O ACTIVITY REPORT COMPLETE**

### Column description

The following is a description of each column printed in the buffer pool report:

#### OMEGAMON XE for DB2 PE Identifiers

The buffer pool report presents data summarized by OMEGAMON XE for DB2 PE identifiers. The report can be ordered by up to three OMEGAMON XE for DB2 PE identifiers. The identifiers used to sort the report are printed in the leftmost column. They are printed whenever they change. The second and third identifiers are indented to appear under the relevant column subheading.

**Note:** Blank or null OMEGAMON XE for DB2 PE identifiers are denoted by the word 'BLANK'.

#### I/O REQUEST

**TOTAL**

The total number of I/O requests.

**AET**

The average elapsed time per I/O request.

#### READ REQUESTS

**TOTAL**

The number of read I/O requests of a specific type.

**TYPE**

The type of read request. The field can contain one of the following values:

- **SYNCH**
  
  Synchronous read request

- **SEQPF**
  
  Sequential prefetch requests (determined at bind time)
**I/O Activity - Detail Report**

- **DYNPF**
  Dynamic prefetch request (triggered at run time by sequential detection logic)

- **LSTPF**
  List prefetch request
  If there are no read requests, a dash (—) is printed.

- **WITH I/O AET**
  The average elapsed time for a read of a specific type.

- **WITH I/O %**
  The percentage of total read requests of a particular type that resulted in an I/O.

- **WITH I/O PAGES/READ**
  Pages read per successful read request of a particular type.

- **W/OUT I/O %**
  The percentage of total read requests of a particular type that did not result in an I/O. This can occur because all the pages requested by a prefetch read were already in the buffer pool.

**WRITE REQUEST**

- **TOTAL**
  The number of write I/O requests of a specific type. Up to two lines are generated, depending on the write type.
  For synchronous writes: count of matching IFCID 008 and 009 record pairs
  For asynchronous writes: count of matching IFCID 009 and 010 record pairs

- **TYPE**
  The type of write request. The field can contain one of the following values:
  - **SYNCH**
    Synchronous write request
  - **ASYNC**
    Asynchronous write request

  If there are no write requests, a dash (—) is printed.

- **CASTOUT**
  The number of synchronous and asynchronous writes due to coupling facility castouts. This field contains one of the following values:
  - **YES**
    The write operations were initiated due to a coupling facility castout.
  - **NO**
    The write operations were initiated as a normal write I/O.

- **AET**
  The average elapsed time per write of a specific type.

- **PAGES/WRITE**
  The number of pages written per write of a specific type.

- **DB OPEN**
  The number of database open requests.
PAGE FAULTS
The number of anticipated page faults. Real storage frames are tested before issuing write.

ACTIVE BUFFERS
The number of active buffers in the pool.

UPDATED PAGES
The number of updated pages in the deferred write queue for the buffer pool that is identified in field QW0008BP or QW0010BP.

SUBTOTAL
When a report is ordered by three identifiers and there is more than one third-level identifier reported under it, a subtotal is printed each time the second-level identifier changes.

TOTAL
When a report is ordered by two or three identifiers and there is more than one second-level identifier reported under it, a total is printed each time the first-level identifier changes.

GRAND TOTAL
A grand total is printed at the end of each location if there is more than one first-level identifier reported.
The EDM pool report provides information about the number of cursor table, package table, or database directory requests, loads from the DASD, their average elapsed times, and the average section lengths of the loaded data items.

Note: The OMEGAMON XE for DB2 PE Statistics reports and traces include EDM pool statistics. Refer to Part 9, “The Statistics Report Set” for more information.

The following command produces the EDM pool report shown in "I/O Activity EDM Pool Report."

```
... I/OACTIVITY
    REPORT
        LEVEL (EDM)
    ...
```

I/O Activity EDM Pool Report

This is an example of an EDM pool report.

| LOCATION / DB / PLAN NAME | TYPE                | CT/PT/DBD REFERENCE | NOT IN EDM POOL | AVERAGE SECTION LENGTH | AVERAGE PER LOAD (BYTES)
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BBDB 10</td>
<td>DATABASE DESCRIPTOR</td>
<td>1</td>
<td>101</td>
<td>0.110000</td>
<td>100.00</td>
</tr>
<tr>
<td>PLAN 01</td>
<td>CURSOR TABLE - HEADER</td>
<td>1</td>
<td>201</td>
<td>0.120000</td>
<td>200.00</td>
</tr>
<tr>
<td>PLAN 02</td>
<td>CURSOR TABLE - DIRECTORY</td>
<td>1</td>
<td>301</td>
<td>0.130000</td>
<td>300.00</td>
</tr>
<tr>
<td>PLAN 03</td>
<td>CURSOR TABLE - RDS SECTION</td>
<td>1</td>
<td>401</td>
<td>0.140000</td>
<td>400.00</td>
</tr>
<tr>
<td>LOCNLOCATIONDB2A COLLCOLLECTIONDB2A PKIDPACKAGEDB2A CTKN'X'C3E3D560C42F2C1'</td>
<td>PACKAGE TABLE - HEADER</td>
<td>1</td>
<td>501</td>
<td>0.150000</td>
<td>500.00</td>
</tr>
<tr>
<td>LOCNSYD2 COLLCOLLECTIONSYD2 PKIDPACKAGESYD2 CTKN'X'C3E3D56DE2E8C4F2'</td>
<td>PACKAGE TABLE - DIRECTORY</td>
<td>1</td>
<td>601</td>
<td>0.160000</td>
<td>600.00</td>
</tr>
<tr>
<td>LOCNLLOCATIONLOCATION1 COLLCOLLECTIONLOCATION1 PKIDPACKAGELOCATION1 PKIDPACKAGELOCATION1 CTKN'X'C3E3D560E2E8C4F1'</td>
<td>PACKAGE TABLE - RDS SECTION</td>
<td>1</td>
<td>701</td>
<td>0.170000</td>
<td>700.00</td>
</tr>
</tbody>
</table>
**Column description**

The following is a description of each column printed in the EDM pool report:

**OMEGAMON XE for DB2 PE Identifiers**

The EDM pool report presents data summarized by OMEGAMON XE for DB2 PE identifiers. The report can be ordered by up to three OMEGAMON XE for DB2 PE identifiers. The identifiers used to sort the report are printed in the leftmost column. They are printed whenever they change. The second and third identifiers are indented to appear under the relevant column subheading.

*Note:* Blank or null OMEGAMON XE for DB2 PE identifiers are denoted by the word 'BLANK'.

**PACKAGE / DBD / PLAN NAME**

The name of the cursor table, package table, or database directory. The package name is printed on four lines, and consists of the following fields:

- **LOCN** Location name
- **COLL** Collection identifier
- **PKID** Package identifier
- **CTKN** Consistency token

**TYPE** The type of data being accessed.

**CT/PT/DBD REFERENCE**

The number of cursor table, package table, or database directory requests performed by the data type specified in the TYPE column.

**NOT IN EDM POOL**

The number of times cursor table, package table, or database directory was not found in the EDM pool. If it is not found in the EDM pool, the request can be satisfied from the buffer pool or the DASD.

**ELAPSED TIME PER LOAD**

The average elapsed time of loads from the buffer pool or DASD.

**AVERAGE SECTION LENGTH (BYTES)**

The average section length of a loaded data item identified in the TYPE column.

**TYPE TOTAL**

When the data contains two or more EDM records with the same data type and the same OMEGAMON XE for DB2 PE identifiers, a type total line is printed with the following heading:

- DBD TOTAL for type DBD
I/O Activity - Detail Report

- PLANNAME TOTAL for type CT
- PACKAGE TOTAL for type PT

**SUBTOTAL**
When a report is ordered by three identifiers and there is more than one third-level identifier reported under it, a subtotal is printed each time the second-level identifier changes.

**TOTAL**
When a report is ordered by two or three identifiers and there is more than one second-level identifier reported under it, a total is printed each time the first-level identifier changes.

**GRAND TOTAL**
A grand total is printed at the end of each location if there is more than one first-level identifier reported.
**Active Log Report**

The active log report provides information about the writing and retrieving of log records. It presents the number of reads, writes, and non-I/O waits related to the active log and the average elapsed times spent waiting for these events.

The report is ordered by the data set identifier.

The following command produces the active log report shown in "I/O Activity Log Report."

---

```
... I/OACTIVITY
    REPORT
    LEVEL (ACTLOG)
...```

---

**I/O Activity Log Report**

This is an example of the I/O Activity log report.

<table>
<thead>
<tr>
<th>INTERVAL</th>
<th>DATASET</th>
<th>I/O REQ</th>
<th>READ REQ</th>
<th>WRITE REQUESTS</th>
<th>OTHER WAITS</th>
<th>TOTAL AET</th>
<th>TOTAL AET</th>
<th>TOTAL CI/WRTE</th>
<th>ALLOC AET</th>
<th>DEALLOC AET</th>
<th>OPEN AET</th>
<th>CLOSE AET</th>
</tr>
</thead>
<tbody>
<tr>
<td>01/30 05:00 - 06/01 05:05</td>
<td>ACTLG001</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>N/C</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.200000</td>
<td>0.200000</td>
</tr>
<tr>
<td>01/30 05:20 - 06/01 05:25</td>
<td>ACTLG002</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>N/C</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.210000</td>
<td>0.210000</td>
</tr>
<tr>
<td>01/30 06:00 - 06/01 06:05</td>
<td>ACTLG003</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>12345.00</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.220000</td>
<td>0.220000</td>
</tr>
<tr>
<td>01/30 07:00 - 06/01 07:05</td>
<td>ACTLG004</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>N/C</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.270000</td>
<td>N/C</td>
</tr>
<tr>
<td>01/30 05:00 - 06/01 05:05</td>
<td>ACTLG005</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>N/C</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.280000</td>
<td>N/C</td>
</tr>
<tr>
<td>01/30 05:00 - 06/01 05:05</td>
<td>ACTLG006</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>N/C</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.290000</td>
<td>N/C</td>
</tr>
<tr>
<td>01/30 05:00 - 06/01 05:05</td>
<td>ACTLG007</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>N/C</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0.300000</td>
<td>N/C</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>01/30 07:00 - 01/30 07:05</td>
<td><strong>14</strong></td>
<td><strong>4</strong></td>
<td><strong>2</strong></td>
<td><strong>12345.00</strong></td>
<td><strong>2</strong></td>
<td><strong>2</strong></td>
<td><strong>2</strong></td>
<td><strong>2</strong></td>
<td><strong>2</strong></td>
<td><strong>2</strong></td>
<td><strong>0.252857</strong></td>
<td><strong>0.255000</strong></td>
</tr>
</tbody>
</table>

---

I/O ACTIVITY REPORT COMPLETE
Column description

The following is a description of each column printed in the active log report:

**OMEGAMON XE for DB2 PE Identifiers**
The active log report presents data summarized by OMEGAMON XE for DB2 PE identifiers. The report can be ordered by up to three OMEGAMON XE for DB2 PE identifiers. The identifiers used to sort the report are printed in the leftmost column. They are printed whenever they change. The second and third identifiers are indented to appear under the relevant column subheading.

**Note:**
1. Blank or null OMEGAMON XE for DB2 PE identifiers are denoted by the word 'BLANK'.
2. **DATASET** is the most significant identifier. It is the 8-byte ID that identifies the active log data set where the reported activity occurs. It has the value ACTLGcxx, where c is the copy number and xx is the sequence number of the active log data set.

**I/O REQ**

<table>
<thead>
<tr>
<th>TOTAL</th>
<th>The total number of I/O requests.</th>
</tr>
</thead>
<tbody>
<tr>
<td>AET</td>
<td>The average elapsed time of all waits.</td>
</tr>
</tbody>
</table>

**READ REQ**

<table>
<thead>
<tr>
<th>TOTAL</th>
<th>The total number of read requests.</th>
</tr>
</thead>
<tbody>
<tr>
<td>AET</td>
<td>The average elapsed time of a read request.</td>
</tr>
</tbody>
</table>

**WRITE REQUESTS**

<table>
<thead>
<tr>
<th>TOTAL</th>
<th>The total number of write requests.</th>
</tr>
</thead>
<tbody>
<tr>
<td>AET</td>
<td>The average elapsed time of a write request.</td>
</tr>
</tbody>
</table>

**CI/WRITE**

| The number of control intervals per write. |

**OTHER WAITS**

<table>
<thead>
<tr>
<th>ALLOC</th>
<th>The number of waits for resource allocation.</th>
</tr>
</thead>
<tbody>
<tr>
<td>AET</td>
<td>The average elapsed time of a wait for resource allocation.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DEALLOC</th>
<th>The number of waits for resource deallocation.</th>
</tr>
</thead>
<tbody>
<tr>
<td>AET</td>
<td>The average elapsed time of waits for resource deallocation.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>OPEN</th>
<th>The number of waits to open data sets.</th>
</tr>
</thead>
<tbody>
<tr>
<td>AET</td>
<td>The average elapsed time of a wait to open data sets.</td>
</tr>
</tbody>
</table>
CLOSE
The number of waits to close data sets.

AET The average elapsed time of a wait to close data sets.

SUBTOTAL
When a report is ordered by three identifiers and there is more than one third-level identifier reported under it, a subtotal is printed each time the second-level identifier changes.

TOTAL
When a report is ordered by two or three identifiers and there is more than one second-level identifier reported under it, a total is printed each time the first-level identifier changes.

GRAND TOTAL
A grand total is printed at the end of each location if there is more than one first-level identifier reported.
Archive Log/BSDS Report

The archive log/BSDS report provides information about the writing of log records and the retrieval of log data. It also contains information about the bootstrap data set that controls the movement of full active log data sets to the archive log.

The report presents the following activity types:

- Archive waits
- Archive read requests
- Archive offload requests
- BSDS read requests
- BSDS write requests

"Archive Log Activity" on page 28-13
This section shows an example of how to produce the Archive Log/BSDS report and explains fields and columns shown in the report.

"Bootstrap Data Set Activity" on page 28-17
This section shows an example of how to produce the Bootstrap Data Set (BSDS) report and explains fields shown in the report.
Archive Log Activity

This section shows an example of how to produce the Archive Log/BSDS report and explains fields and columns shown in the report.

The following command produces the Archive Log/BSDS report in "I/O Activity—Archive Log Activity Report."

```
... 
LOGIN ACTIVITY 
REPORT 
LEVEL (ARCLOG) ... 
```

I/O Activity—Archive Log Activity Report

This is an example of an archive log/BSDS report.

```
LOCATION: LOCATION1 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-1
GROUP: GROUP1 I/O ACTIVITY REPORT - ARCLOG REQUESTED FROM: NOT SPECIFIED
MEMBER: DB1A TO: NOT SPECIFIED
SUBSYSTEM: DB1A ORDER: DATASET-INTERVAL INTERVAL FROM: 01/30/10 14:00:00.00 TO: 01/30/10 14:35:00.00

ARCHIVE LOG ACTIVITY -----------------------------------------------------------------------------------------------
DATASET INTERVAL WAIT TYPE TOTAL AET OTHER WAITS TOTAL AET
----------------------------------------------- -------------- -------- --------- -------------------------- -------- --------
00111583 01/30 14:25 - 01/30 14:30 READ FROM DASD 0 N/C ALLOCATE 0 N/C
READ FROM TAPE 0 N/C DEALLOCATE 0 N/C
OFFLOAD 1 71.230139 OPEN 0 N/C
OTHER 0 N/C CLOSE 0 N/C
HSM RECALL 0 N/C
BLOCKS/OFFLOAD 26999.00 
CATALOG LOCATE 0 N/C
MULTI DATA SET TAPE VOLUME 0 N/C
TAPE VOLUME POSITIONING 0 N/C
WTOR ISSUED 0 N/C
DATA SET UNAVAILABLE 0 N/C
PHYSICAL UNIT UNAVAILABLE 0 N/C
READER SERVICE UNAVAILABLE 0 N/C
00111584 01/30 14:30 - 01/30 14:35 READ FROM DASD 0 N/C ALLOCATE 0 N/C
READ FROM TAPE 0 N/C DEALLOCATE 0 N/C
OFFLOAD 1 67.210716 OPEN 0 N/C
OTHER 0 N/C CLOSE 0 N/C
HSM RECALL 0 N/C
BLOCKS/OFFLOAD 26999.00 
CATALOG LOCATE 0 N/C
MULTI DATA SET TAPE VOLUME 0 N/C
TAPE VOLUME POSITIONING 0 N/C
WTOR ISSUED 0 N/C
DATA SET UNAVAILABLE 0 N/C
PHYSICAL UNIT UNAVAILABLE 0 N/C
READER SERVICE UNAVAILABLE 0 N/C
00111585 01/30 14:30 - 01/30 14:35 READ FROM DASD 0 N/C ALLOCATE 0 N/C
READ FROM TAPE 0 N/C DEALLOCATE 0 N/C
OFFLOAD 1 64.683949 OPEN 0 N/C
OTHER 0 N/C CLOSE 0 N/C
HSM RECALL 0 N/C
BLOCKS/OFFLOAD 26999.00 
CATALOG LOCATE 0 N/C
MULTI DATA SET TAPE VOLUME 0 N/C
TAPE VOLUME POSITIONING 0 N/C
WTOR ISSUED 0 N/C
DATA SET UNAVAILABLE 0 N/C
PHYSICAL UNIT UNAVAILABLE 0 N/C
READER SERVICE UNAVAILABLE 0 N/C
ARCHIVE LOG ACTIVITY -----------------------------------------------------------------------------------------------
LOCATION: LOCATION1 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-2
GROUP: GROUP1 I/O ACTIVITY REPORT - ARCLOG REQUESTED FROM: NOT SPECIFIED
MEMBER: DB1A TO: NOT SPECIFIED
SUBSYSTEM: DB1A ORDER: DATASET-INTERVAL INTERVAL FROM: 01/30/10 14:00:00.00 TO: 01/30/10 14:35:00.00

ARCHIVE LOG ACTIVITY -----------------------------------------------------------------------------------------------

Chapter 28. I/O Activity Detail Reports 28-13
## I/O Activity - Detail Report

<table>
<thead>
<tr>
<th>DATASET INTERVAL</th>
<th>WAIT TYPE</th>
<th>TOTAL AET</th>
<th>OTHER WAITS</th>
<th>TOTAL AET</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>** TOTAL **</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### 10111583
01/30 14:25 - 01/30 14:30
- **READ FROM DASD** 0 N/C ALLOCATE 2 0.119761
- **READ FROM TAPE** 0 N/C DEALLOCATE 2 0.000664
- **OFFLOAD** 0 N/C OPEN 2 0.005564
- **OTHER** 8 0.038971
- **CATALOG LOCATE** 0 N/C
- **MULTI DATA SET TAPE VOLUME** 0 N/C
- **TAPE VOLUME POSITIONING** 0 N/C
- **WTOR ISSUED** 0 N/C
- **DATA SET UNAVAILABLE** 0 N/C
- **PHYSICAL UNIT UNAVAILABLE** 0 N/C
- **READER SERVICE UNAVAILABLE** 0 N/C

### 10111584
01/30 14:25 - 01/30 14:30
- **READ FROM DASD** 0 N/C ALLOCATE 2 0.108854
- **READ FROM TAPE** 0 N/C DEALLOCATE 2 0.000614
- **OFFLOAD** 0 N/C OPEN 2 0.006097
- **OTHER** 5 0.049937
- **CATALOG LOCATE** 0 N/C
- **MULTI DATA SET TAPE VOLUME** 0 N/C
- **TAPE VOLUME POSITIONING** 0 N/C
- **WTOR ISSUED** 0 N/C
- **DATA SET UNAVAILABLE** 0 N/C
- **PHYSICAL UNIT UNAVAILABLE** 0 N/C
- **READER SERVICE UNAVAILABLE** 0 N/C

### 10111585
01/30 14:30 - 01/30 14:35
- **READ FROM DASD** 0 N/C ALLOCATE 2 0.061112
- **READ FROM TAPE** 0 N/C DEALLOCATE 2 0.006672
- **OFFLOAD** 0 N/C OPEN 2 0.003976
- **OTHER** 8 0.021197
- **CATALOG LOCATE** 0 N/C
- **MULTI DATA SET TAPE VOLUME** 0 N/C
- **TAPE VOLUME POSITIONING** 0 N/C
- **WTOR ISSUED** 0 N/C
- **DATA SET UNAVAILABLE** 0 N/C
- **PHYSICAL UNIT UNAVAILABLE** 0 N/C
- **READER SERVICE UNAVAILABLE** 0 N/C

### ARCHIVE LOG ACTIVITY

<table>
<thead>
<tr>
<th>DATASET INTERVAL</th>
<th>WAIT TYPE</th>
<th>TOTAL AET</th>
<th>OTHER WAITS</th>
<th>TOTAL AET</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>** TOTAL **</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### 10111584
01/30 14:25 - 01/30 14:30
- **READ FROM DASD** 0 N/C ALLOCATE 2 0.108854
- **READ FROM TAPE** 0 N/C DEALLOCATE 2 0.000614
- **OFFLOAD** 0 N/C OPEN 2 0.006097
- **OTHER** 5 0.049937
- **CATALOG LOCATE** 0 N/C
- **MULTI DATA SET TAPE VOLUME** 0 N/C
- **TAPE VOLUME POSITIONING** 0 N/C
- **WTOR ISSUED** 0 N/C
- **DATA SET UNAVAILABLE** 0 N/C
- **PHYSICAL UNIT UNAVAILABLE** 0 N/C
- **READER SERVICE UNAVAILABLE** 0 N/C
### Column description

The following is a description of each column printed in the archive log activity section of the archive log/BDSD report.

**OMEGAMON XE for DB2 PE Identifiers**

The archive log activity report presents data summarized by OMEGAMON XE for DB2 PE identifiers. The report can be ordered by up to three OMEGAMON XE for DB2 PE identifiers. The identifiers used to sort the report are printed in the leftmost column. They are printed whenever they change. The second and third identifiers are indented to appear under the relevant column subheading.

**Note:**

1. Blank or null OMEGAMON XE for DB2 PE identifiers are denoted by the word 'BLANK'.
2. **DATASET** is the most significant identifier. It is the 8-byte ID that identifies the archive log data set where the reported activity occurs. It has the value `cxxxxxxx`, where `c` is the copy number and `xxxxxxx` is the sequence number. The sequence number is the same as the last seven characters of the data set name.

**WAIT TYPE**

**READ FROM DASD**

The total number of waits and the average elapsed time of a wait for archive log reads from the DASD.

**READ FROM TAPE**

The total number of waits and the average elapsed time of a wait of archive log reads from the tape.

**OFFLOAD**

The total number of waits for archive log write requests and the average elapsed time of waits per archive log write.

**OTHER**

The total number of non-I/O waits and the average elapsed time of non-I/O waits on the archive log data set.

**OTHER WAITS**

The following fields identify the other waits section.

**ALLOCATE**

The total number of waits and the average elapsed time of a wait for resource allocation.
I/O Activity - Detail Report

DEALLOCATE
The total number of waits and the average elapsed time of a wait for resource deallocation.

OPEN
The total number of waits and the average elapsed time of a wait to open a data set.

CLOSE
The total number of waits and the average elapsed time of a wait to close a data set.

HSM RECALL
The total number of waits and the average elapsed time of a wait for HSM to recall data sets.

CATALOG LOCATE
The total number of waits and the average elapsed time of a wait to locate data sets through the catalog.

MULTI DATA SET TAPE VOLUME
The total number of waits and the average elapsed time per wait for multi-data set tape volume.

TAPE VOLUME POSITIONING
The total number of waits and the average elapsed time per wait for tape volume positioning.

WTOR ISSUED
The total number of waits and the average elapsed time of waits due to a write-to-operator message being issued.

DATA SET UNAVAILABLE
The total number of waits and the average elapsed time of a wait due to a data set being unavailable.

PHYSICAL UNIT UNAVAILABLE
The total number of waits and the average elapsed time of a wait due to an unavailable physical unit.

READER SERVICE UNAVAILABLE
The total number of waits and the average elapsed time per wait for an unavailable reader service task.

SUBTOTAL
When a report is ordered by three identifiers and there is more than one third-level identifier reported under it, a subtotal is printed each time the second-level identifier changes.

TOTAL
When a report is ordered by two or three identifiers and there is more than one second-level identifier reported under it, a total is printed each time the first-level identifier changes.

GRAND TOTAL
A grand total is printed at the end of each location if there is more than one first-level identifier reported.
I/O Activity - Bootstrap Data Set

Bootstrap Data Set Activity

This section shows an example of how to produce the Bootstrap Data Set (BDS) report and explains fields shown in the report.

The following command produces the archive log/BDS report in "I/O Activity—Bootstrap Data Set Activity".

```plaintext
... 
IOACTIVITY
REPORT
   LEVEL (ARCLOG) 
...
```

I/O Activity—Bootstrap Data Set Activity

<table>
<thead>
<tr>
<th>DATASET</th>
<th>INTERVAL</th>
<th>WAIT TYPE</th>
<th>TOTAL</th>
<th>AET</th>
</tr>
</thead>
<tbody>
<tr>
<td>BSDS0001</td>
<td>01/29 14:00 - 01/29 14:05</td>
<td>READ</td>
<td>2</td>
<td>0.001947</td>
</tr>
<tr>
<td>BSDS0001</td>
<td>01/29 14:00 - 01/29 14:05</td>
<td>WRITE</td>
<td>2</td>
<td>0.001098</td>
</tr>
<tr>
<td>BSDS0001</td>
<td>01/29 14:00 - 01/29 14:05</td>
<td>READ AND WRITE</td>
<td>4</td>
<td>0.001523</td>
</tr>
<tr>
<td>BSDS0001</td>
<td>01/29 14:20 - 01/29 14:25</td>
<td>READ</td>
<td>94</td>
<td>0.001947</td>
</tr>
<tr>
<td>BSDS0001</td>
<td>01/29 14:20 - 01/29 14:25</td>
<td>WRITE</td>
<td>94</td>
<td>0.001098</td>
</tr>
<tr>
<td>BSDS0001</td>
<td>01/29 14:20 - 01/29 14:25</td>
<td>READ AND WRITE</td>
<td>188</td>
<td>0.001390</td>
</tr>
<tr>
<td>BSDS0001</td>
<td>01/29 14:30 - 01/29 14:35</td>
<td>READ</td>
<td>460</td>
<td>0.001233</td>
</tr>
<tr>
<td>BSDS0001</td>
<td>01/29 14:30 - 01/29 14:35</td>
<td>WRITE</td>
<td>212</td>
<td>0.000902</td>
</tr>
<tr>
<td>BSDS0001</td>
<td>01/29 14:30 - 01/29 14:35</td>
<td>READ AND WRITE</td>
<td>672</td>
<td>0.001157</td>
</tr>
</tbody>
</table>

** TOTAL **

<table>
<thead>
<tr>
<th>DATASET</th>
<th>INTERVAL</th>
<th>WAIT TYPE</th>
<th>TOTAL</th>
<th>AET</th>
</tr>
</thead>
<tbody>
<tr>
<td>BSDS0001</td>
<td>01/29 14:00 - 01/29 14:05</td>
<td>READ</td>
<td>1264</td>
<td>0.001233</td>
</tr>
<tr>
<td>BSDS0001</td>
<td>01/29 14:00 - 01/29 14:05</td>
<td>WRITE</td>
<td>531</td>
<td>0.000902</td>
</tr>
<tr>
<td>BSDS0001</td>
<td>01/29 14:00 - 01/29 14:05</td>
<td>READ AND WRITE</td>
<td>1795</td>
<td>0.001206</td>
</tr>
<tr>
<td>BSDS0002</td>
<td>01/29 14:00 - 01/29 14:05</td>
<td>READ</td>
<td>2</td>
<td>0.001660</td>
</tr>
<tr>
<td>BSDS0002</td>
<td>01/29 14:00 - 01/29 14:05</td>
<td>WRITE</td>
<td>2</td>
<td>0.001049</td>
</tr>
<tr>
<td>BSDS0002</td>
<td>01/29 14:00 - 01/29 14:05</td>
<td>READ AND WRITE</td>
<td>4</td>
<td>0.001354</td>
</tr>
</tbody>
</table>

*** GRAND TOTAL ***

<table>
<thead>
<tr>
<th>DATASET</th>
<th>INTERVAL</th>
<th>WAIT TYPE</th>
<th>TOTAL</th>
<th>AET</th>
</tr>
</thead>
<tbody>
<tr>
<td>BSDS0001</td>
<td>01/29 14:00 - 01/29 14:05</td>
<td>READ</td>
<td>1795</td>
<td>0.001392</td>
</tr>
<tr>
<td>BSDS0001</td>
<td>01/29 14:00 - 01/29 14:05</td>
<td>WRITE</td>
<td>1062</td>
<td>0.001049</td>
</tr>
<tr>
<td>BSDS0001</td>
<td>01/29 14:00 - 01/29 14:05</td>
<td>READ AND WRITE</td>
<td>2857</td>
<td>0.001261</td>
</tr>
</tbody>
</table>

Field description

The following field descriptions are for the bootstrap data set activity section of the archive log/BDS report.

OMEGAMON XE for DB2 PE Identifiers

The bootstrap data set activity report presents data summarized by OMEGAMON XE for DB2 PE identifiers. The report can be ordered by up to three OMEGAMON XE for DB2 PE identifiers. The identifiers used to sort the report are printed in the leftmost column. They are printed...
whenever they change. The second and third identifiers are indented to appear under the relevant column subheading.

Note:
1. Blank or null OMEGAMON XE for DB2 PE identifiers are denoted by the word ‘BLANK’.
2. **DATASET** is the most significant identifier. It is the 8-byte ID that identifies the BSDS data set where the reported activity occurs. It can have either of the following values:
   - BSDS0001
   - BSDS0002

**WAIT TYPE**

**READ** The total number of BSDS reads and the average elapsed time per BSDS read.

**WRITE**

The total number of BSDS writes and the average elapsed time per BSDS write.

**READ AND WRITE**

The total number of BSDS reads and writes and the average elapsed time per BSDS read and write.

**SUBTOTAL**

When a report is ordered by three identifiers and there is more than one third-level identifier reported under it, a subtotal is printed each time the second-level identifier changes.

**TOTAL**

When a report is ordered by two or three identifiers and there is more than one second-level identifier reported under it, a total is printed each time the first-level identifier changes.

**GRAND TOTAL**

A grand total is printed at the end of each location if there is more than one first-level identifier reported.
Cross-Invalidation Report

The cross-invalidation report presents buffer refresh events due to cross invalidation summarized by selected OMEGAMON XE for DB2 PE identifiers. If two DB2 systems compete for read/write interest on a page set or partition, a certain amount of buffer cross-invalidation activity occurs to maintain DB2 buffer pool coherency between the two systems.

Cross-invalidation (XI) renders a higher percentage of the buffer pool data invalid. It has the effect of reducing the buffer pool size and thus the buffer pool hit ratio. Buffer pool pages must be continually refreshed when high cross-invalidation levels are reached. This can be a significant overhead in data sharing if workloads between DB2 systems are not properly balanced.

The following command produces the cross-invalidation report in "I/O Activity Cross-Invalidation Report."

```
IOACTIVITY
REPORT
  LEVEL (XI)
```

I/O Activity Cross-Invalidation Report

| INTERVAL       | SYNCHRONOUS READS | -SEQUENTIAL PREFETCHES-
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>GPPOOL DASD</td>
<td>GPPOOL DASD</td>
</tr>
<tr>
<td>01/30 18:50 - 01/30 18:55</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AUTH_10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLAN_10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DB=4 OB=2</td>
<td>PIECE=#0</td>
<td>PAGE=#X'00000002' BPID=BP22</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>01/30 18:55 - 01/30 19:00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AUTH_10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLAN_10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DB=4 OB=2</td>
<td>PIECE=#0</td>
<td>PAGE=#X'00000002' BPID=BP22</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>01/30 19:00 - 01/30 19:05</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AUTH_10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLAN_10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DB=4 OB=2</td>
<td>PIECE=#0</td>
<td>PAGE=#X'00000002' BPID=BP22</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PAGE=#X'00000004' BPID=BP22</td>
</tr>
<tr>
<td></td>
<td>SUM OF PLAN_10*</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>01/30 19:05 - 01/30 19:10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AUTH_10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLAN_10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DB=4 OB=2</td>
<td>PIECE=#0</td>
<td>PAGE=#X'00000002' BPID=BP22</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PAGE=#X'00000004' BPID=BP22</td>
</tr>
<tr>
<td></td>
<td>SUM OF PLAN_10*</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*** GRAND TOTAL ***

19 | 4 | 2 | 0

I/O ACTIVITY REPORT COMPLETE

Column description

The following is a description of each column printed in the cross-invalidation report:
OMEGAMON XE for DB2 PE Identifiers

The XI report presents data summarized by OMEGAMON XE for DB2 PE identifiers. The report can be ordered by up to three OMEGAMON XE for DB2 PE identifiers. The identifiers used to sort the report are printed in the leftmost column. They are printed whenever they change. The second and third identifiers are indented to appear under the relevant column subheading.

Note: Blank or null OMEGAMON XE for DB2 PE identifiers are denoted by the word 'BLANK'.

PAGE
The name of the page involved in the cross invalidation. The name consists of the following parts:
- **DB**: Database name
- **OB**: Page set name
- **PIECE#**: Page set piece number
- **PAGE#**: Page number
- **BPID**: Buffer pool ID

If DATABASE or PAGESET are selected in the ORDER option, DB or OB are not shown as part of the page name.

SYNCHRONOUS READS
The number of times the page was refreshed via a synchronous read for a particular combination of OMEGAMON XE for DB2 PE identifiers and cross-invalidated page:
- **GBPOOL**: From the group buffer pool.
- **DASD**: From DASD.

SEQUENTIAL PREFETCHES
The number of times the page was refreshed via a sequential prefetch for a particular combination of OMEGAMON XE for DB2 PE identifiers and cross-invalidated page:
- **GBPOOL**: From the group buffer pool.
- **DASD**: From DASD.

SUM OF
The totals for all pages within a combination of OMEGAMON XE for DB2 PE identifiers if two or more entries are printed in the PAGE column.

SUBTOTAL
When a report is ordered by three identifiers and there is more than one third-level identifier reported under it, a subtotal is printed each time the second-level identifier changes.

TOTAL
When a report is ordered by two or three identifiers and there is more than one second-level identifier reported under it, a total is printed each time the first-level identifier changes.
I/O Activity - Cross-Invalidation Report

GRAND TOTAL
A grand total is printed at the end of each location if there is more than one first-level identifier reported.
Accounting Report
Part 6. The Locking Report Set

These topics provide information about the Locking reports.

For an introduction to the Locking report set and general locking information refer to the Reporting User's Guide. It also provides information on input to locking reports.

Chapter 29, “Member-Scope Traces and Reports,” on page 29-1
Member-scope traces present events in chronological sequence within the DB2 subsystem (member) where the events occurred, whereas reports show these events aggregated by the OMEGAMON XE for DB2 PE identifiers you have specified.

Chapter 30, “Group-Scope Traces and Reports,” on page 30-1
In group-scope traces, events are reported in a chronological sequence within the DB2 data sharing group, regardless of which member of the group actually generated the events.

Chapter 31, “Identifiers Used in Locking,” on page 31-1
Chapter 32, “The Locking Header of Reports and Traces,” on page 32-1
This topic describes the headers and fields of Locking reports and traces. The report header and trace header are similar for all reports and traces. All other report or trace sections differ depending on the type of report and are described in the respective report or trace topics.

Chapter 33, “Locking Activity Report,” on page 33-1
Here you find a detailed description about Locking activity reports.

Chapter 34, “Locking Trace,” on page 34-1
Chapter 35, “The Locking File Data Set,” on page 35-1
The locking file data set creates a sequential data set of formatted DB2 locking detail records that can be loaded into the OMEGAMON XE for DB2 PE performance database using the DB2 load utility.
Chapter 29. Member-Scope Traces and Reports

Member-scope traces present events in chronological sequence within the DB2 subsystem (member) where the events occurred, whereas reports show these events aggregated by the OMEGAMON XE for DB2 PE identifiers you have specified.

OMEGAMON XE for DB2 PE can present data from several DB2 members within a data sharing group. The data in member-scope reports is presented by a combination of location, group, subsystem, and member. Whenever one of the values changes, a new page is started and the page number is initialized.

The information in this section is only applicable to DB2 data sharing environments.

Note: For an introduction to the Locking report set and general locking information refer to the Reporting User's Guide

Member-Scope Locking Trace

This is an example of a Member-Scope Locking trace.

LOCATION: PMODBZ1 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-1
GROUP: DBZ1 LOCKING TRACE - DEADLOCK
MEMBER: SZ11 REQUESTED FROM: NOT SPECIFIED
SUBSYSTEM: SZ11 ACTUAL FROM: 06/04/10 08:10:31.30
DB2 VERSION: V10 TO: NOT SPECIFIED
PRIMAUTH CONNNAME CONNTYPE
ORIGAUTH CORRNAME CORRID
PLANNAME CONNECT RELATED TIMESTAMP EVENT TYPE NAME EVENT SPECIFIC DATA
------------------------------ ----------------- -------- --------- ----------------------- ----------------------------------------
SKA java DRDA 08:10:31.30440975 DEADLOCK COUNTER =11975 WAITERS = 2
counter =11975 waiters = 2
SKA 'BLANK' C614015B874D N/P TSTAMP =06/04/10 08:10:31.28
TSTAMP =06/04/10 08:10:31.28
DISTSERV SERVER HASH =''0000000000012021' HASH =''0000000000012021'
REDLOC :::FFFF:9.152.78. OB =32 --------- BLOCKER is HOLDER -->VICTIM--
REDLOC :::FFFF:9.152.78. OB =32
PRIMAUTH =SZ11 CONNECT =SERVER LUW=G9984E80.P64B.C614015B874D
MEMBER =SZ11 CONNECT =SERVER LUW=G9984E80.P64A.C614015B8746
PLANNAMe=DISTSERV CORRID =java REQUEST =CHANGE WORTH = 10
PLANNAMe=DISTSERV CORRID =java STATE =X STMTINFO=DYNAMIC
REQUEST =CHANGE WORTH = 10
ENDUSER =ska STATE =X STMTINFO=DYNAMIC
ENDUSER =ska
WSNAME =mupfel ENDUSER =ska
TRANSAC=java PROGNAME=SYSSH200 WSNAMe =mupfel
TRANSAC=java PROGNAME=SYSSH200
COLLID =NULLID PROGNAME=SYSSH200
LOCATION=N/P COLLID =NULLID
CONTOKEN=X'5359534C564L301' COLLID =NULLID
STMTID =X'0000000000012099' STMTID =X'0000000000012099'
---------------- WAITER ---------------- STMTID =X'0000000000012099'
LUW=G9984E80.P64A.C614015B874D LOCATION=N/P
MEMBER =SZ11 LOCATION=N/P
CONTOKEN=X'5359534C564L301'
STMTID =X'0000000000012099'
© Copyright IBM Corp. 1985, 2013
Locking Activity - Member-Scope

SKA 'BLANK' C614015B874D OB =32
DISTSERV SERVER
REGLOC ::FFFF:9.152.78.
ENDUSER :ska
WSNAME :mupfel
TRANSACT:java

------------ BLOCKER IS WAITER ------------
LUW=G9984E80.P648.C614015B8346
MEMBER =S21I CONNECT =SERVER
PLANNAME=DISTSERV CORRID =java
DURATION=COMMIT PRIMAUTH=SKA
STATE =X STMTINFO=DYNAMIC
ENDUSER =ska
WSNAME =mupfel
TRANSAC=java
PRGNAME=SYSSH200
COLLID =NULLID
LOCATION=N/P
CONTOKEN =X'5359534C564C3031'
STMTID =X'00000000000012B9'

---------------- WAITER ------------*
LUW=G9984E80.P64A.C614015B874D
MEMBER =SZ11 CONNECT =SERVER
PLANNAME=DISTSERV CORRID =java
DURATION=COMMIT PRIMAUTH=SKA
REQUEST =CHANGE WORTH = 17
STATE =X STMTINFO=DYNAMIC
ENDUSER =ska
WSNAME =mupfel
TRANSAC=java
PRGNAME=SYSSH200
COLLID =NULLID
LOCATION=N/P
CONTOKEN =X'5359534C564C3031'
STMTID =X'00000000000012B9'

----------------- W A I T E R -----------------
LUW=G9984E80.P64A.C614015B8346
MEMBER =SZ11 CONNECT =SERVER
PLANNAME=DISTSERV CORRID =java
DURATION=COMMIT PRIMAUTH=SKA
REQUEST =CHANGE WORTH = 18
STATE =X STMTINFO=DYNAMIC
ENDUSER =ska
WSNAME =mupfel
TRANSAC=java
PRGNAME=SYSSH200
COLLID =NULLID
LOCATION=N/P
CONTOKEN =X'5359534C564C3031'
STMTID =X'00000000000012B9'

LOCATION: PMODBZ1 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0)
GROUP: DBZ1
MEMBER: S21I
SUBSYSTEM: S21I
DB VERSION: V10
DB2 PERFORMANCE EXPERT DB2 PERFORMANCE EXPERT (V5R2M0)
LOCATION: PMODBZ1 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0)
GROUP: DBZ1
MEMBER: S21I
SUBSYSTEM: S21I
DB VERSION: V10

--- LOCK RESOURCES ---
PLANNAME CONNECT RELATED TIMESTAMP EVENT TYPE NAME EVENT SPECIFIC DATA
SKA java ORA: 10:30.28514203 DEADLOCK
SKA 'BLANK' C614015B7C7C N/P
DISTSERV SERVER TABLE DB =TDKDB HASH =X'00012011'
REGLOC ::FFFF:9.152.78. OB =32
ENDUSER :ska
WSNAME :mupfel
TRANSACT:java

--------- BLOCKER IS HOLDER ---------
LUW=G9984E80.P649.C614015B7C7C
MEMBER =S21I CONNECT =SERVER
PLANNAME=DISTSERV CORRID =java
DURATION=COMMIT PRIMAUTH=SKA
STATE =X STMTINFO=DYNAMIC
ENDUSER =ska
WSNAME =mupfel
TRANSAC=java
PRGNAME=SYSSH200
COLLID =NULLID
LOCATION=N/P
CONTOKEN =X'5359534C564C3031'
STMTID =X'00000000000012B9'

--- LOCK RESOURCES ---
PLANNAME CONNECT RELATED TIMESTAMP EVENT TYPE NAME EVENT SPECIFIC DATA
SKA java ORA: 10:30.28514203 DEADLOCK
SKA 'BLANK' C614015B7C7C N/P
DISTSERV SERVER TABLE DB =TDKDB HASH =X'00012011'
REGLOC ::FFFF:9.152.78. OB =32
ENDUSER :ska
WSNAME :mupfel
TRANSACT:java

--------- BLOCKER IS WAITER ---------
LUW=G9984E80.P648.C614015B8346
MEMBER =S21I CONNECT =SERVER
PLANNAME=DISTSERV CORRID =java
DURATION=COMMIT PRIMAUTH=SKA
STATE =X STMTINFO=DYNAMIC
ENDUSER =ska
WSNAME =mupfel
TRANSAC=java
PRGNAME=SYSSH200
COLLID =NULLID
LOCATION=N/P
CONTOKEN =X'5359534C564C3031'
STMTID =X'00000000000012B9'

--- LOCK RESOURCES ---
PLANNAME CONNECT RELATED TIMESTAMP EVENT TYPE NAME EVENT SPECIFIC DATA
SKA java ORA: 10:30.28514203 DEADLOCK
SKA 'BLANK' C614015B7C7C N/P
DISTSERV SERVER TABLE DB =TDKDB HASH =X'00012011'
REGLOC ::FFFF:9.152.78. OB =32
ENDUSER :ska
WSNAME :mupfel
TRANSACT:java

--------- BLOCKER IS WAITER ---------
LUW=G9984E80.P649.C614015B7C7C
MEMBER =S21I CONNECT =SERVER
PLANNAME=DISTSERV CORRID =java
DURATION=COMMIT PRIMAUTH=SKA
STATE =X STMTINFO=DYNAMIC
ENDUSER =ska
WSNAME =mupfel
TRANSAC=java
PRGNAME=SYSSH200
COLLID =NULLID
LOCATION=N/P
CONTOKEN =X'5359534C564C3031'
STMTID =X'00000000000012B9'
### Locking Activity - Member-Scope

**PROGNAME**: SYSSH200  
**COLLID**: NULLID  
**LOCATION**: N/P  
**CONTOKEN**: X'5359534C564C3031'  
**STMTID**: X'00000000000012B9'

------------ WAITER -----------
**LUW**: G9984E80.P649.C614015B7C7C
**MEMBER**: SZ11  
**CONNECT**: SERVER
**PLANNAME**: DISTSERV  
**CORRID**: java  
**DURATION**: COMMIT  
**PRIMAUTH**: SKA
**REQUEST**: CHANGE  
**WORTH**: 17  
**STATE**: X  
**STMTINFO**: DYNAMIC
**ENDUSER**: ska  
**WSNAME**: mupfel  
**TRANSACT**: java

**LOCATION**: PMODBZ1  
**GROUP**: DBZ1  
**MEMBER**: SZ11  
**SUBSYSTEM**: SZ11  
**DB2 VERSION**: V10  
**SCOPE**: MEMBER

**PRIMAUTH CORRNAME CONNTYPE**
**ORIGAUTH CORRNMBR INSTANCE**
**PLANNAME CONNECT RELATED TIMESTAMP EVENT TYPE NAME EVENT SPECIFIC DATA**

<table>
<thead>
<tr>
<th>Event Type</th>
<th>Name</th>
<th>Event Specific Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEADLOCK</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

------------ LOCK RESOURCE -----------
**HASH**: X'00002011'  
**BLOCKER is HOLDER -----------
**LUW**: G9984E80.P649.C614015B7C7C
**MEMBER**: SZ11  
**CONNECT**: SERVER
**PLANNAME**: DISTSERV  
**CORRID**: java  
**DURATION**: COMMIT  
**PRIMAUTH**: SKA
**REQUEST**: CHANGE  
**WORTH**: 17  
**STATE**: X  
**STMTINFO**: DYNAMIC
**ENDUSER**: ska  
**WSNAME**: mupfel  
**TRANSACT**: java

**LOCATION**: PMODBZ1  
**GROUP**: DBZ1  
**MEMBER**: SZ11  
**SUBSYSTEM**: SZ11  
**DB2 VERSION**: V10  
**SCOPE**: MEMBER

**PRIMAUTH CORRNAME CONNTYPE**
**ORIGAUTH CORRNMBR INSTANCE**
**PLANNAME CONNECT RELATED TIMESTAMP EVENT TYPE NAME EVENT SPECIFIC DATA**

<table>
<thead>
<tr>
<th>Event Type</th>
<th>Name</th>
<th>Event Specific Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEADLOCK</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

------------ LOCK RESOURCE -----------
**HASH**: X'00002011'  
**BLOCKER is WAITER -----------
**LUW**: G9984E80.P649.C614015B7C7C
**MEMBER**: SZ11  
**CONNECT**: SERVER
**PLANNAME**: DISTSERV  
**CORRID**: java  
**DURATION**: COMMIT  
**PRIMAUTH**: SKA
**REQUEST**: CHANGE  
**WORTH**: 17  
**STATE**: X  
**STMTINFO**: DYNAMIC
**ENDUSER**: ska  
**WSNAME**: mupfel  
**TRANSACT**: java

---

Chapter 29. Member-Scope Traces and Reports  29-3
Locking Activity - Member-Scope

LOCATION: PM0DB21 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-7
GROUP: DB21 LOCKING TRACE - DEADLOCK
MEMBER: S211 REQUESTED FROM: NOT SPECIFIED
SUBSYSTEM: S211 ACTUAL FROM: 06/04/10 08:10:31.30
DB2 VERSION: V10 SCOPE: MEMBER
PRIMAUTH CORRNAME CONNTYPE ORIGAUTH CORRNMBR INSTANCE
PLANNANE CONNNAME EVENT TIMESTAMP RELATED TIMESTAMP EVENT TYPE NAME EVENT SPECIFIC DATA
------------------------------ ----------------- -------- --------- ----------------------- ----------------------------------------
S211 java DRDA 06:04:10 08:10:31.31
DISTSERV SERVER TABLE DB =TDKDB HASH =X'00012011'
MEMBER =S211 CONNECT =SERVER
PLANNANE=DISTSERV CORRID =java STATE =S STMTINFO=DYNAMIC
ENDUSER =ska TRANSACT=java DURATION=COMMIT PRIMAUTH=SKA REQUEST =CHANGE WORTH = 18
STATE =X STMTINFO=DYNAMIC
ENDUSER =ska
WSNAME =mupfel
TRANSACT:java

LOCATION: PM0DB21 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-8
GROUP: DB21 LOCKING TRACE - DEADLOCK
MEMBER: S211 REQUESTED FROM: NOT SPECIFIED
SUBSYSTEM: S211 ACTUAL FROM: 06/04/10 08:10:31.30
DB2 VERSION: V10 SCOPE: MEMBER
PRIMAUTH CORRNAME CONNTYPE ORIGAUTH CORRNMBR INSTANCE
PLANNANE CONNNAME EVENT TIMESTAMP RELATED TIMESTAMP EVENT TYPE NAME EVENT SPECIFIC DATA
------------------------------ ----------------- -------- --------- ----------------------- ----------------------------------------
S211 java DRDA 06:04:10 08:10:31.31
DISTSERV SERVER TABLE DB =TDKDB HASH =X'00012011'
MEMBER =S211 CONNECT =SERVER
PLANNANE=DISTSERV CORRID =java STATE =S STMTINFO=DYNAMIC
ENDUSER =ska TRANSACT=java DURATION=COMMIT PRIMAUTH=SKA REQUEST =CHANGE WORTH = 18
STATE =X STMTINFO=DYNAMIC
ENDUSER =ska
WSNAME =mupfel
TRANSACT:java
Locking Activity - Member-Scope
### Locking Activity - Member-Scope

<table>
<thead>
<tr>
<th>PRIMAUTH</th>
<th>CORRNAME</th>
<th>CONNTYPE</th>
<th>ORIGAUTH</th>
<th>CORRNMBR</th>
<th>INSTANCE</th>
<th>EVENT</th>
<th>TIMESTAMP</th>
<th>---</th>
<th>LOCK RESOURCE ---</th>
<th>EVENT SPECIFIC DATA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TABLE</td>
<td>DB = TDKDB</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>DB = 32</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TABLE</td>
<td>DB = TDKDB</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>DB = 32</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TABLE</td>
<td>DB = TDKDB</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>DB = 32</td>
<td></td>
</tr>
</tbody>
</table>
Locking Activity - Member-Scope

LOCATION: PMODBZ1
GROUP: DB2
MEMBER: S211
SUBSYSTEM: S211
DB2 VERSION: V10
PRIMAUTH CORRNAME CONNTYPE
ORIGAUTH CORRNAME INSTANCE
PLANNAME CONNECT RELATED TIMESTAMP EVENT

--- LOCK RESOURCE ---

EVENT SPECIFIC DATA

<table>
<thead>
<tr>
<th>SKA</th>
<th>java</th>
<th>DRDA</th>
<th>C61401B7C7C</th>
<th>LOCATION: PMODBZ1</th>
<th>GROUP: DB2</th>
<th>MEMBER: S211</th>
<th>SUBSYSTEM: S211</th>
<th>DB2 VERSION: V10</th>
<th>PRIMAUTH CORRNAME</th>
<th>CONNTYPE</th>
</tr>
</thead>
</table>

LOCKING TRACE COMPLETE
Accounting Report
Chapter 30. Group-Scope Traces and Reports

In group-scope traces, events are reported in a chronological sequence within the DB2 data sharing group, regardless of which member of the group actually generated the events.

The member name is printed in the body of the trace for each reported event, so that it is easy to see the member where the event occurred. Similarly, group-scope reports show events that are aggregated by the OMEGAMON XE for DB2 PE identifiers you specified. Data in group-scope reports is presented by member.

The information in this section is only applicable to DB2 data sharing environments.

Note: For an introduction to the Locking report set and general locking information refer to the Reporting User's Guide.

Group-Scope Locking Report

This is an example of a Group-Scope Locking report.
## Locking Activity - Group-Scope

<table>
<thead>
<tr>
<th></th>
<th>AUS00801</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>TOTAL</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>8.314863</td>
<td>0</td>
<td>N/C</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GRAND TOTAL</td>
<td>16</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>16</td>
<td>2.080090</td>
<td>0</td>
<td>N/C</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>16</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Chapter 31. Identifiers Used in Locking

Note: For an introduction to the Locking report set and general locking information refer to the Reporting User’s Guide.

In addition to the standard OMEGAMON XE for DB2 PE identifiers, Locking reports and traces use two other identifiers to show the type of resource and event type:

**RESOURCETYPE - Resource type**
- The type of lock resource. You can specify one of the values shown in Table 33-1 on page 33-3.

**TYPE - Event type**
- Specifies which event types are to be included in, or excluded from, the lock detail trace. The valid values for this field are shown in Table 33-1.

<table>
<thead>
<tr>
<th>Type</th>
<th>Events</th>
</tr>
</thead>
<tbody>
<tr>
<td>IRLMREQ</td>
<td>Lock, unlock, change, query, and notify requests</td>
</tr>
<tr>
<td>CLAIMREQ</td>
<td>Claim acquire, claim change, and claim release</td>
</tr>
<tr>
<td>DRAINREQ</td>
<td>Drain request and drain release</td>
</tr>
<tr>
<td>PLOCKREQ</td>
<td>Page set or partition as well as page P-Lock requests</td>
</tr>
<tr>
<td>IRLMSUSP</td>
<td>The beginning of lock, unlock, change, query, and notify suspensions</td>
</tr>
<tr>
<td>DRAINSUSP</td>
<td>The beginning of drain suspensions</td>
</tr>
<tr>
<td>LATCHSUSP</td>
<td>The beginning of page latch suspensions</td>
</tr>
<tr>
<td>IRLMRES</td>
<td>The end (resumption) of lock, unlock, change, query, and notify suspensions</td>
</tr>
<tr>
<td>DRAINRES</td>
<td>The end (resumption) of drain suspensions</td>
</tr>
<tr>
<td>LATCHRES</td>
<td>The end (resumption) of page latch suspensions</td>
</tr>
<tr>
<td>TIMEOUT</td>
<td>Timeouts</td>
</tr>
<tr>
<td>DEADLOCK</td>
<td>Deadlocks</td>
</tr>
<tr>
<td>LOCKSUMMARY</td>
<td>Lock summary events</td>
</tr>
<tr>
<td>LOCKAVOID</td>
<td>Successful lock avoidance events</td>
</tr>
</tbody>
</table>

Table 33-1. Event Types

The default is all event types.

Note: TYPE can also be used with the REDUCE and FILE subcommands of locking. These subcommands support a limited number of types, as follows:
- Valid types for REDUCE are: IRLMRES, DRAINRES, and LATCHRES.
- Valid types for FILE are: IRLMREQ, CLAIMREQ, DRAINREQ, and LOCKAVOID.

If a non-valid type for REDUCE or FILE is used with EXCLUDE, the event type is not filtered.
Locking Activity - Report Identifiers

If no valid types for REDUCE or FILE are used with INCLUDE, an empty report or file is produced.
Chapter 32. The Locking Header of Reports and Traces

This topic describes the headers and fields of Locking reports and traces. The report header and trace header are similar for all reports and traces. All other report or trace sections differ depending on the type of report and are described in the respective report or trace topics.

Note: For an introduction to the Locking report set and general locking information refer to the Reporting User’s Guide.

Layout of a Locking Report Header

The following example shows the layout of a report header, where the letter $x$ is a placeholder marking the maximum size of the data section of each field.

```
LOCATION: xxxxxxxxxxxxxxxx OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: l-n
GROUP: xxxxxxxx LOCKING REPORT - xxxxxxxxx REQUESTED FROM: mm/dd/yy hh:mm:ss.nn
MEMBER: xxxxxxxx TO: mm/dd/yy hh:mm:ss.nn
SUBSYSTEM: xxxx ORDER: xxxxxx INTERVAL FROM: mm/dd/yy hh:mm:ss.nn
DB2 VERSION: Vn Rn SCOPE: xxxxxx TO: mm/dd/yy hh:mm:ss.nn
```

Layout of the Locking Trace Header

All traces have the same layout. This example shows the layout of a trace header, where the letter $x$ is a placeholder marking the maximum size of the data section of each field. In this example the trace is ordered by the event timestamp.

```
LOCATION: xxxxxxxxxxxxxxxx OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-n
GROUP: xxxxxxxx REQUESTED FROM: mm/dd/yy hh:mm:ss.nn
MEMBER: xxxxxxxx INTERVAL FROM: mm/dd/yy hh:mm:ss.nn
SUBSYSTEM: xxxx ACTUAL FROM: mm/dd/yy hh:mm:ss.nn
DB2 VERSION: Vn Rn PAGE DATE: mm/dd/yy
```

Field descriptions of Locking headers

Headings are printed on all reports and traces at the start of each page. Locking reports and traces carry the following header information:

- **LOCATION**
  The DB2 reporting location. If the location name is not available, the DB2 data sharing group name is printed in this field. If the DB2 data sharing group name does not exist, the DB2 subsystem ID is printed.

- **GROUP**
  The name of the DB2 data sharing group. This field shows N/A if there is no group name.

- **MEMBER**
  The name of the DB2 data sharing member or the member name of the DB2 subsystem. This field shows N/A if there is no member name.

  This field is not printed on the report page showing the GROUP BUFFERPOOLS PARAMETERS.

- **SUBSYSTEM**
  The ID of the DB2 subsystem that generated the data. This field is not printed on the report page showing the GROUP BUFFERPOOLS PARAMETERS.

- **DB2 VERSION**
  The DB2 version number of the subsystem that generated the data.

© Copyright IBM Corp. 1985, 2013
OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (VnRnMn)

The product name and the version, release, and modification level.

REPORT or TRACE type

For report, this can be:
- SUSPENSION
- LOCKOUT
- DETAIL

For trace, this can be:
- DEADLOCK
- TIMEOUT
- SUSPENSION
- LOCKOUT
- DETAIL

ORDER

If the ORDER option of the REPORT or TRACE subcommand was used to arrange the report entries, the selected keywords are shown in this field. Depending on the context, the OMEGAMON XE for DB2 PE identifiers by which lock events are grouped are shown here.

SCOPE

Scope of the report or trace, this can be MEMBER or GROUP. A member-scope report or trace shows data from a group for each individual member. In a group-scope report or trace, the data from individual members is consolidated and presented for the entire group.

PAGE

The page number in the format lll-nnnnnn, where lll denotes the location number within the report and nnnnnn the page number within the location.

REQUESTED FROM and TO

The FROM and TO dates and times specified in the REPORT or TRACE subcommand.

If both FROM and TO dates and times are omitted from the REPORT subcommand, the FROM and TO dates and times specified in GLOBAL are printed. If only the FROM date and time or only the TO date and time has been specified, NOT SPECIFIED is printed for the unspecified value.

If FROM and TO are not specified in REPORT or GLOBAL, NOT SPECIFIED appears for both the FROM and TO values.

If you have specified FROM and TO times without dates in REPORT or GLOBAL, ALL DATES is printed along with the specified times.

INTERVAL FROM

The start date and time of the first reduction interval covered by the report. If REDUCE is not specified, the INTERVAL defaults to 0 and the timestamps of the first and last events are printed.

INTERVAL TO

The end date and time of the last reduction interval covered by the report. If REDUCE is not specified, the INTERVAL defaults to 0 and the timestamps of the first and last events are printed.

ACTUAL FROM/TO

The date and time of the first and last record included in the log for a location, group, subsystem, or member.
PAGE DATE
The date of the timestamps printed on this page. A page break occurs at
the change of the date. This is useful if a trace page contains more than
one entry and the date is not shown for each entry.
Chapter 33. Locking Activity Report

Here you find a detailed description about Locking activity reports.

Note: For an introduction to the Locking Activity report set and general locking information refer to the Reporting User’s Guide

“Lock Suspension Report” on page 33-2
The Lock Suspension report summarizes all Lock Suspension activities across a specified time period.

“Using Lock Suspension Data with Spreadsheets” on page 33-9
This section provides the necessary information to enable you to use lock suspension report data that is created with the LOCKING REPORT SPREADSHEETDD command in a spreadsheet program.

“Lockout Report” on page 33-12
The lockout report summarizes timeouts and deadlocks occurring within a specified period of time.

“Locking Detail Report” on page 33-15
The Locking Detail (also referred to as Lock Detail) report is based on IFCID 21, which records the detail lock requests.
The Lock Suspension report summarizes all Lock Suspension activities across a specified time period.

The suspensions are reported by any combination of up to three OMEGAMON XE for DB2 PE identifiers. The report summarizes the Lock Suspension activities of:

- An IRLM request (except when the resource type is a drain lock).
- An IRLM request where the resource type is a drain lock.
- A drain request where the claim count is not zero.
  This suspension occurs when the agent making the drain request has to wait for the claim count on the particular resource to become zero.
- A suspension of a page latch request.
  This suspension occurs when the agent making the page latch request has to wait for a page which is currently being held by another agent.

The Lock Suspension report is produced if level SUSPENSION is specified in the REPORT subcommand. The ORDER subcommand specifies by which OMEGAMON XE for DB2 PE identifiers the report is to be sorted.

Optionally, the SPREADSHEETDD subcommand option can be used to create a data set with Lock Suspension data that can be imported in spreadsheet programs for individual analyses. Refer to “Using Lock Suspension Data with Spreadsheets” on page 33-9 for more details.

Every suspension results in a normal resume or a lockout (deadlock or timeout), or is canceled (in the case of page latch suspensions). In any case, and if the suspension delay is unacceptable, review the plans and associated tables and indexes.

“Layout of a Suspension Report” shows the layout of a suspension report. The letter x is a placeholder marking the maximum size of a field. See “Lock Suspension Report” on page 33-8 for an example of a suspension report.

The report presents data summarized by OMEGAMON XE for DB2 PE identifiers. The report can be sorted by up to three identifiers. For group-scope reports, the member name is added implicitly as an additional identifier and sort criterion.

**Layout of a Suspension Report**

This is the layout of a suspension report.

```
LOCATION: xxxxxxxxxxxxxxxx OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-n
GROUP: xxxxxxxx LOCKING REPORT - SUSPENSION REQUESTED FROM: mm/dd/yy hh:mm:ss.nn
MEMBER: xxxxxxxx ORDER: xxxxxx INTERVAL FROM: mm/dd/yy hh:mm:ss.nn
SUBSYSTEM: xxxx SCOPE: xxxxxx TO: mm/dd/yy hh:mm:ss.nn
IDENT1xx
IDENT2xx --- LOCK RESOURCE --- TOTAL LOCAL GLOB. S.NFY --- NORMAL --- TIMEOUT/CANCEL --- DEADLOCK ---
MEMBER TYPE NAME SUSPENS LATCH IRLMQ OTHER NMBR AET NMBR AET NMBR AET NMBR AET
xxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Locking Activity - Suspension Report
```

33-2 OMEGAMON XE for DB2 PE & PM: Report Reference
Field description

The header information to this report is described in Chapter 32, “The Locking Header of Reports and Traces,” on page 32-1.

**IDENT1XX, IDENT2XX, IDENT3XX**

These mark the positions where the order criteria are listed and reported. In group-scope reports, MEMBER is automatically added as the second, third, or fourth identifier.

If you specify DATABASE, PAGESET, or both, in the ORDER option, the database name, page set name, or both names, are removed from the resource name. In this case, 'BLANK' is printed.

**MEMBER**

Group member name of the DB2 subsystem. This is only printed for member-scope reports.

**LOCK RESOURCE TYPE**

The type of resource on which the suspended request is made. Valid values are shown in Table 33-1.

**Table 33-1. Lock Resource Type**

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTERBUF</td>
<td>Alter buffer pool lock</td>
</tr>
<tr>
<td>BINDLOCK</td>
<td>Autobind lock and remote bind lock for the serialization of local autobinds or packages, remote binds, and remote rebinds of packages</td>
</tr>
<tr>
<td>CATM CAT</td>
<td>CATMAINT convert catalog lock</td>
</tr>
<tr>
<td>CATM DIR</td>
<td>CATMAINT convert directory lock</td>
</tr>
<tr>
<td>CATM MIG</td>
<td>CATMAINT migration lock</td>
</tr>
<tr>
<td>CDB PLCK</td>
<td>DDF communications database P-lock</td>
</tr>
<tr>
<td>COLLECT</td>
<td>Collection ID</td>
</tr>
<tr>
<td>DATABASE</td>
<td>Locking of the DBD</td>
</tr>
<tr>
<td>DATAPAGE</td>
<td>Data page locking</td>
</tr>
<tr>
<td>DBALLOC</td>
<td>Start and stop lock on the database allocation table</td>
</tr>
<tr>
<td>DBCMD SER</td>
<td>Database command serialization</td>
</tr>
<tr>
<td>DBD</td>
<td>DBD load lock</td>
</tr>
<tr>
<td>DBD PLCK</td>
<td>DBD P-lock</td>
</tr>
<tr>
<td>DRAIN</td>
<td>All types of drain locking</td>
</tr>
<tr>
<td>DRAIN CS</td>
<td>Cursor stability drain lock</td>
</tr>
<tr>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>DRAIN RR</td>
<td>Repeatable read drain lock</td>
</tr>
<tr>
<td>DRAIN W</td>
<td>Write drain lock</td>
</tr>
<tr>
<td>EXCP UPD</td>
<td>Database group exception update lock</td>
</tr>
<tr>
<td>GBP CAST</td>
<td>Group buffer pool level castout P-lock</td>
</tr>
<tr>
<td>GBP S/S</td>
<td>Group buffer pool start and stop lock</td>
</tr>
<tr>
<td>HASH-ANC</td>
<td>Hash anchor lock</td>
</tr>
<tr>
<td>HDRPHASHB</td>
<td>BACKUP SYSTEM or RESTORE SYSTEM utility lock</td>
</tr>
<tr>
<td>INDEX KEY</td>
<td>Index Key lock</td>
</tr>
<tr>
<td>INDEXEOF</td>
<td>Index end-of-file lock</td>
</tr>
<tr>
<td>INDEXPAGE</td>
<td>Index page locking</td>
</tr>
<tr>
<td>LOB</td>
<td>Large object</td>
</tr>
<tr>
<td>LPL/GREC</td>
<td>Database group exception LPL/GRECP lock</td>
</tr>
<tr>
<td>LPLRECVRY</td>
<td>Logical page list recovery</td>
</tr>
<tr>
<td>MASSDEL</td>
<td>Mass delete lock</td>
</tr>
<tr>
<td>OPENLOCK</td>
<td>Page set or data set open lock</td>
</tr>
<tr>
<td>OTHER</td>
<td>All unlisted resource types</td>
</tr>
<tr>
<td>P/P CAST</td>
<td>Page set and partition level castout P-lock</td>
</tr>
<tr>
<td>P/P PLCK</td>
<td>Page set and partition P-lock</td>
</tr>
<tr>
<td>PAGE</td>
<td>Resource involved in page latch suspensions</td>
</tr>
<tr>
<td>PAGEPLCK</td>
<td>Page P-lock</td>
</tr>
<tr>
<td>PAGESET</td>
<td>Nonpartitioned table spaces and indexes. Drained at the page set level.</td>
</tr>
<tr>
<td>PART NSPL</td>
<td>Partitions of partitioned table spaces and indexes using the non-SPL (selective partition locking) scheme.</td>
</tr>
<tr>
<td>PART SPL</td>
<td>Partitions of partitioned table spaces and indexes using the SPL (selective partition locking) scheme.</td>
</tr>
<tr>
<td>RLF PLCK</td>
<td>RLF P-lock</td>
</tr>
<tr>
<td>ROW</td>
<td>Data row locking</td>
</tr>
<tr>
<td>RPR_DBDB</td>
<td>Repair DBD test and diagnose lock</td>
</tr>
<tr>
<td>SCA ACCS</td>
<td>SCA access for restart or redo information</td>
</tr>
<tr>
<td>SKCT</td>
<td>Skeleton cursor table locking</td>
</tr>
</tbody>
</table>
Table 33-1. Lock Resource Type (continued)

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SKPT</td>
<td>Skeleton package table</td>
</tr>
<tr>
<td></td>
<td>Note, if the lock resource type has a value of SKPT the Lock Detail report shows compressed parts of the SKPT resource name as hexadecimal strings in reports and traces. It consists of the following parts:</td>
</tr>
<tr>
<td></td>
<td>• Collection ID, which is compressed (18 bytes)</td>
</tr>
<tr>
<td></td>
<td>• Program name, which is compressed (8 bytes)</td>
</tr>
<tr>
<td></td>
<td>• Consistency token (8 bytes)</td>
</tr>
<tr>
<td></td>
<td>With the hexadecimal value you can compare values of different locks. See &quot;Layout of a Locking Detail report for the lock resource type SKPT&quot; on page 33-15 for an example of a Locking Detail report for the lock resource type SKPT.</td>
</tr>
<tr>
<td></td>
<td>SKPT resource names are shown for the Locking Activity report at the following LEVEL:</td>
</tr>
<tr>
<td></td>
<td>• LOCKOUT</td>
</tr>
<tr>
<td></td>
<td>• SUSPENSION</td>
</tr>
<tr>
<td></td>
<td>SKPT resource names are shown for the Locking Activity trace at the following LEVEL:</td>
</tr>
<tr>
<td></td>
<td>• LOCKOUT</td>
</tr>
<tr>
<td></td>
<td>• TIMEOUT</td>
</tr>
<tr>
<td></td>
<td>• SUSPENSION</td>
</tr>
<tr>
<td></td>
<td>• DETAIL</td>
</tr>
</tbody>
</table>

SYSLGRNG Buffer manager SYSLGRNG recording lock
TABLE Table locking
TREEPLCK Index tree P-lock
UTIL EXC Utility exclusive execution lock
UTIL UID Utility UID lock
UTILSER Utility serialization lock
XML LOCK XML lock

Note: For a suspended request where the resource type is not supplied, N/P is printed.

LOCK RESOURCE NAME
The name on which the suspended request is made. Each part of the lock resource name is printed on a separate line. The abbreviations shown in the report are explained, in alphabetical order, in Table 33-2.

Table 33-2. Lock Resource Name Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANCH</td>
<td>Anchor point ID</td>
</tr>
<tr>
<td>BPID</td>
<td>Buffer pool ID</td>
</tr>
<tr>
<td>COLL</td>
<td>Collection name</td>
</tr>
<tr>
<td>CKTN</td>
<td>Consistency token</td>
</tr>
</tbody>
</table>
Locking Activity - Suspension Report

Table 33-2. Lock Resource Name Abbreviations (continued)

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB</td>
<td>Database name</td>
</tr>
<tr>
<td>HASH</td>
<td>Database group exception hash class</td>
</tr>
<tr>
<td>OB</td>
<td>Object name</td>
</tr>
<tr>
<td>PAGE</td>
<td>Physical page</td>
</tr>
<tr>
<td>PART</td>
<td>Partition</td>
</tr>
<tr>
<td>PKID</td>
<td>Package name</td>
</tr>
<tr>
<td>PLAN</td>
<td>Plan name</td>
</tr>
<tr>
<td>RMID</td>
<td>Resource manager ID</td>
</tr>
<tr>
<td>ROW</td>
<td>Data row</td>
</tr>
<tr>
<td>ROWI</td>
<td>Row ID for LOB</td>
</tr>
<tr>
<td>SUBP</td>
<td>Subpage</td>
</tr>
<tr>
<td>UID</td>
<td>Utility ID</td>
</tr>
<tr>
<td>VER#</td>
<td>Version number of LOB</td>
</tr>
</tbody>
</table>

Note:
1. The database names and object names are translations obtained from the IFCID 105 and 107 records. If these records are not available, the decimal representation of the database and object names are printed.
2. If you specify DATABASE, PAGESET, or both, in the ORDER option, the database name, page set name, or both names, are removed from the resource name and printed in the OMEGAMON XE for DB2 PE identifier column. If the name only consists of the database and page set, 'N/P' is printed in the resource name column. If the resource name does not contain the database and page set, 'BLANK' is printed in the OMEGAMON XE for DB2 PE identifier column and all resource name parts are printed in the lock resource block.

TOTAL SUSPENDS
The number of suspensions for the particular combination of OMEGAMON XE for DB2 PE identifiers.

SUSPEND REASONS
The reason why a particular request was suspended. The requests composing the particular combination of OMEGAMON XE for DB2 PE identifiers and lock resource can be suspended for several reasons. The SUSPEND REASONS column shows all reasons identified by the IRLM resume records. Therefore, the sum of the counts in this column can differ from the TOTAL SUSPENDS count.

The categorized reasons for suspension are:

LOCAL
Local resource contention. This occurs when you request access to a local resource that is locked.

LATCH
IRLM latch contention. This occurs when the IRLM needs to serialize a resource. For example, the IRLM serializes the adding
and removing of locks to the lock table. The lock table is latched for a short period of time, and the resulting suspensions, if any, are brief.

GLOB.
Global contention. This occurs when you request access to a global resource that is locked.

IRLMQ
IRLM queued request.

S.NFY
Intersystem message sending.

OTHER
Suspensions other than those listed here. Suspensions reported as OTHER are either serviceability values, drain suspensions, contentions with retained locks, or page latch suspensions.

RESUME REASONS
The reasons for resumption of the suspended tasks. The reason can be normal, timeout, deadlock, and canceled (canceled only applies to page latch suspensions).

NORMAL NMBR
The number of suspensions that ended when the task resumed normal processing after completion of the lock request. In page latch suspensions, this is the number of suspensions where the latch requester was not canceled.

NORMAL AET
The average elapsed time of a suspension that ended in the task resuming normally. In page latch suspensions, this is the average elapsed time of a suspension where the latch requester was not canceled.

TIMEOUT NMBR
The number of waits to access locked resources that resulted in exceeding a preset time interval.

TIMEOUT AET
The average elapsed time of a resumption due to a timeout.

CANCEL NMBR
The number of page latch suspensions that ended with the latch requester being canceled.

CANCEL AET
The average elapsed time of a page latch suspension that ended with the latch requester being canceled.

DEADLOCK NMBR
The number of deadlocks.

DEADLOCK AET
The average duration of a deadlock.
SUM OF
The sum printed for the lowest-level identifier when there is more than one combination of request type, resource type, and lock resource reported under it.

GROUP TOTAL
The sum of report entries that belong to a data sharing group if more than one member of the group is reported for a particular combination of OMEGAMON XE for DB2 PE identifiers. A GROUP TOTAL only appears in group-scope reports.

SUBTOTAL
When a report is ordered by three identifiers and there is more than one third-level identifier reported under it, a subtotal is printed each time the second-level identifier changes.

TOTAL
When a report is ordered by two or three identifiers and there is more than one second-level identifier reported under it, a total is printed each time the first-level identifier changes.

GRAND TOTAL
If there is more than one first-level identifier reported, a grand total is printed at the end of each group in a group-scope report or at the end of each member in a member-scope report.

Lock Suspension Report

The following sample suspension report is produced with this command:

```
; LOCKING
  REPORT
;```

This is a sample Lock Suspension report.

```
LOCATION: OMPDA21 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-1
GROUP: N/P LOCKING REPORT - SUSPENSION REQUESTED FROM: NOT SPECIFIED
MEMBER: N/P TO: NOT SPECIFIED
SUBSYSTEM: DA21 ORDER: PRIMAUTH-PLANNAME INTERVAL FROM: 02/14/11 19:46:09.24
DB2 VERSION: V10 SCOPE: MEMBER TO: 02/14/11 19:48:51.36

--SUSPEND REASONS-- ----------RESUME REASONS -----------
PRIMAUTH -- -LOCK RESOURCE- - - TOTAL LOCAL GLOB. S.NFY ---- NORMAL ---- TIMEOUT/CANCEL --- DEADLOCK ---
PLANNAME TYPE NAME SUSPENDS LATCH IRLMQ OTHER NMBR AET NMBR AET NMBR AET
------------------ --------- ----------------------- -------- ----- ----- ----- ---- ----------- ---- ----------- ---- -----------
SKPT 15 0 0 0 15 0.000042 0 N/C 0 N/C
1500
COLL(HEX)= X'112233445566778899001122334455667788'
PKID(HEX)= X'1122334455667788'

... LOCKING REPORT COMPLETE
```
Using Lock Suspension Data with Spreadsheets

This section provides the necessary information to enable you to use lock suspension report data that is created with the LOCKING REPORT SPREADSHEETDD command in a spreadsheet program.

It is assumed that you created a data set with lock suspension data on the host by using the SPREADSHEETDD option of the LOCKING REPORT command. Refer to the Report Command Reference for more information about the SPREADSHEETDD option. Further, it is assumed that you downloaded the data set to your client as a text file (choose ascii or text, not binary, as transfer type in your file transfer program). The data set should be available as a plain text file in ASCII format on your client.

It is assumed that you are familiar with the use of spreadsheet programs. Modern spreadsheets provide means to import data from plain text files, provided that data in these files is organized as records and individual fields of the records are separated by a known separator character. The file containing lock suspension data has its fields separated by colons (;). Therefore, you must specify the colon as the separator character (also called a delimiter) when you are importing the data into the spreadsheet program. See the help information of the spreadsheet of your choice for details on how to import data.

After the data is imported to your spreadsheet, the records from the plain text file are represented in spreadsheet rows and the fields are represented in spreadsheet columns.

The remainder of this section describes details about the data as it is initially represented in the spreadsheet. Further use, such as sorting, filtering, analysis, and interpretation is not described. For the latter, see "Lock Suspension Report" on page 33-2.

- The first row contains report header information, similar to OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) Locking Report Suspension.

You might notice how the colon-separated format in the plain text file converts to subsequent cells in a spreadsheet row.

- The second row contains the column labels, as shown in Table 33-3. Approximately 35 columns are shown; the precise number depends on what was specified with the ORDER subcommand option when the data was generated.

- The third and all following rows contain the accumulated lock suspension data values.

Empty cells represent missing data values, usually shown in reports as N/A, N/C, or N/P.

Table 33-3. Spreadsheet representation of lock suspension data

<table>
<thead>
<tr>
<th>Col.</th>
<th>Column label</th>
<th>Includes the following DB2 lock resource types</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Location</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Group</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Member</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Subsystem</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Database</td>
<td></td>
</tr>
</tbody>
</table>
### Locking Activity - Suspension Report

#### Table 33-3. Spreadsheet representation of lock suspension data (continued)

<table>
<thead>
<tr>
<th>Col.</th>
<th>Column label</th>
<th>Includes the following DB2 lock resource types</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>Pageset</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>(Content varies)</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Suspensions Total Occurrences</td>
<td>All</td>
</tr>
<tr>
<td>9</td>
<td>Suspensions Average Elapsed Time</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Deadlocks Total Occurrences</td>
<td>All</td>
</tr>
<tr>
<td>11</td>
<td>Timeouts Total Occurrences</td>
<td>All</td>
</tr>
<tr>
<td>12</td>
<td>Row Lock Suspension Occurrences</td>
<td>Row (X'18')</td>
</tr>
<tr>
<td>13</td>
<td>Row Lock Suspension Average Elapsed Time</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>Page Lock Suspension Occurrences</td>
<td>Datapage (X'00')</td>
</tr>
<tr>
<td>15</td>
<td>Page Lock Suspension Average Elapsed Time</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>Page Latch Suspension Occurrences</td>
<td>Number of IFCID 226 and 227 pairs</td>
</tr>
<tr>
<td>17</td>
<td>Page Latch Suspension Average Elapsed Time</td>
<td></td>
</tr>
</tbody>
</table>
| 18   | Pageset Lock Suspension Occurrences | • Pageset (X'02')  
|      |                                     | • Partitioned table space (X'03')           |
|      |                                     | • Partition (X'06')                           |
| 19   | Pageset Lock Suspension Average Elapsed Time |                                               |
| 20   | Database Lock Suspension Occurrences| Database (X'01')                              |
| 21   | Database Lock Suspension Average Elapsed Time |                                               |
| 22   | Table Lock Suspension Occurrences   | Table (X'10')                                 |
| 23   | Table Lock Suspension Average Elapsed Time |                                               |
| 24   | LOB Lock Suspension Occurrences     | LOB (X'30')                                   |
| 25   | LOB Lock Suspension Average Elapsed Time |                                               |
| 26   | Drain Lock Suspension Occurrences   | • CS-read drain (X'14')  
|      |                                     | • RR-read drain (X'15')  
|      |                                     | • Write drain (X'16')                        |
| 27   | Drain Lock Suspension Average Elapsed Time |                                               |
| 28   | Page P-Lock Suspension Occurrences  | Page P-Lock (X'1E')                           |
| 29   | Page P-Lock Suspension Average Elapsed Time |                                               |
| 30   | Pageset P-Lock Suspension Occurrences| Pageset/partition P-Lock (X'1D')             |
| 31   | Pageset P-Lock Suspension Average Elapsed Time |                                               |
### Locking Activity - Suspension Report

**Table 33-3. Spreadsheet representation of lock suspension data (continued)**

<table>
<thead>
<tr>
<th>Col.</th>
<th>Column label</th>
<th>Includes the following DB2 lock resource types</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>Other P-Lock Suspension Occurrences</td>
<td>• Index manager tree P-Lock (X'1C')</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• DDF CDB P-Lock (X'1F')</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Group Buffer Pool level castout</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Pageset or partition level castout</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• P-LOCK (X'20')</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Pageset or partition level castout</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• RLF P-LOCK (X'22')</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• DBD P-LOCK (X'23')</td>
</tr>
<tr>
<td>33</td>
<td>Other P-Lock Suspension Average</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Elapsed Time</td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>Miscellaneous Lock Suspension</td>
<td>All others not listed above.</td>
</tr>
<tr>
<td></td>
<td>Occurrences</td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>Miscellaneous Lock Suspension</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Average Elapsed Time</td>
<td></td>
</tr>
</tbody>
</table>
Lockout Report

The lockout report summarizes timeouts and deadlocks occurring within a specified period of time.

The report shows the number of times an agent, identified by up to three OMEGAMON XE for DB2 PE identifiers, has been timed out or involved in a deadlock when requesting a particular resource. In addition, it shows the other contenders for the resource and the number of times they act as holders or waiters.

There is no correlation between the number of deadlock events reported by Locking reports and traces and the number of deadlocks reported in Accounting and Statistics reports. Whereas Accounting and Statistics reports count all deadlock occurrences, regardless of how they resolve, Locking reports only those deadlocks that were resolved by DB2. DB2 can resolve a deadlock either by making a process roll back, thereby releasing the locks it holds on resources, or by requesting a process to terminate.

The lockout report is produced if level LOCKOUT is specified in the REPORT subcommand and if there is at least one combination of a lockout agent's identifier satisfying the FROM and TO, and INCLUDE or EXCLUDE criteria.

The ORDER subcommand specifies by which OMEGAMON XE for DB2 PE identifiers the report is to be sorted. You can specify up to three identifiers.

**Note:** For an introduction to the Locking report set and general locking information refer to the [Reporting User's Guide](#).

Layout of a Lockout Report

This is the layout of a lockout report. The letter x is a placeholder marking the maximum size of a field. See “Example of a Lockout Report” on page 33-14 for an example of a lockout report.
Field description

Here is a description of all fields except for the report header, which is described in Chapter 32, “The Locking Header of Reports and Traces,” on page 32-1.

LOCK RESOURCE TYPE
The type of resource involved in the lockout. Valid values are shown in Table 33-1 on page 33-3

LOCK RESOURCE NAME
The name of the resource on which the timeout or deadlock occurred. Each part of the lock resource name is printed on a separate line. The abbreviations shown in the report are explained, in alphabetical order, in Table 33-2 on page 33-5

TIMEOUTS
The number of times the resource was involved in a timeout.

DEADLOCKS
The number of times the resource was involved in a deadlock.

There is no correlation between the number of deadlocks reported by Locking reports and traces and the number of deadlocks reported in Accounting and Statistics reports. Whereas Accounting and Statistics reports count all deadlock occurrences, regardless of how they resolve, Locking reports only those deadlocks that were resolved by DB2. DB2 can resolve a deadlock either by making a process roll back, thereby releasing the locks on resources, or by requesting a process to terminate.

AGENTS
The agents in contention for the resource during the lockout. This block consists of the following columns:

MEMBER
The agent's member name. In a non-data-sharing environment, this field shows N/P.

PLANNNAME
The agent's plan name or the word SYSTEM if there is contention with a retained lock.

CONNECT
The agent's connection name.

CORRID
The agent's correlation identifier.

BLOCKER/HOLDER
For timeouts, the number of times the agent held the resource during the lockout.
For deadlocks, the number of times the agent was the blocker, either as a holder or a waiter.

WAITER
The number of times the agent waited for the resource during the lockout.

LOCKOUTS FOR
The number of timeout and deadlock records aggregated for the currently reported set of OMEGAMON XE for DB2 PE identifiers.

For timeouts, this value is equivalent to the sum of the entries in the TIMEOUTS column.
Locking Activity - Lockout Report

A deadlock record involves several resources. Therefore, this value differs from the sum of the entries in the DEADLOCKS column.

GROUP TOTAL
The sum of report entries that belong to a data sharing group if more than one member of the group is reported for a particular combination of the DB2 identifiers. A GROUP TOTAL only appears in group-scope reports.

SUBTOTAL
When a report is ordered by three identifiers and there is more than one third-level identifier reported under it, a subtotal is printed each time the second-level identifier changes.

TOTAL
When a report is ordered by two or three identifiers and there is more than one second-level identifier reported under it, a total is printed each time the first-level identifier changes.

GRAND TOTAL
If there is more than one first-level identifier reported, a grand total is printed at the end of each group in a group-scope report or at the end of each member in a member-scope report.

Example of a Lockout Report

The following command is used to produce the example of the Lockout report:

```
... LOCKING REPORT LEVEL (LOCKOUT) ...
```

Here is the example of the Lockout report:

LOCATION: STLECI OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-1
GROUP: N/P LOCKING REPORT - LOCKOUT REQUESTED FROM: NOT SPECIFIED
MEMBER: N/P TO: NOT SPECIFIED
SUBSYSTEM: VA1A ORDER: PRIMAUTH-PLANNAME INTERVAL FROM: 06/04/09 00:15:44.20
DB2 VERSION: V10 SCOPE: MEMBER TO: 06/04/09 00:15:44.20

<table>
<thead>
<tr>
<th>PRIMAUTH</th>
<th>LOCK RESOURCE</th>
<th>LOCKED RESOURCE</th>
<th>RESOURCE</th>
<th>TIMEOUTS</th>
<th>DEADLOCKS</th>
<th>AGENTS</th>
<th>BLOCKER/</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYSADM</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

CTKN=000000000000000000000000
COLL(HEX)= X'112233445566778899001122334455667788'
PRTD(HEX)= X'1122334455667788'
** LOCKOUTS FOR DSNTEP3 ** 1 0

LOCKING REPORT COMPLETE
Locking Activity - Lock Detail Report

Locking Detail Report

The Locking Detail (also referred to as Lock Detail) report is based on IFCID 21, which records the detail lock requests.

The Locking Detail report is produced if level DETAIL is specified in the REPORT subcommand.

The ORDER subcommand specifies by which OMEGAMON XE for DB2 PE identifiers the report is to be sorted. In this sample the data is accumulated and ordered by DATABASE-PAGESET.

“Layout of a Locking Detail Report” shows the layout of a Locking Detail report. The letter x is a placeholder marking the maximum size of a field. See “Locking Detail Report” on page 33-17 for an example of a Locking Detail report.

Note: For an introduction to the Locking report set and general locking information refer to the Reporting User’s Guide.

Layout of a Locking Detail Report

Note: If the lock resource type has a value of SKPT (skeleton package table locking) the Lock Detail report shows compressed parts of the SKPT resource name as hexadecimal strings in reports and traces. It consists of the following parts:
• Collection ID, which is compressed.
• Program name, which is compressed.
• Consistency token

With the hexadecimal value you can compare values of different locks. See “Layout of a Locking Detail report for the lock resource type SKPT” for an example of a Locking Detail report for the lock resource type SKPT.

Layout of a Locking Detail report for the lock resource type SKPT

Here is an example of the layout of a Locking Detail report for the lock resource type SKPT.
### Field description

Here is a description of all fields except for the report header, which is described in [Chapter 32, “The Locking Header of Reports and Traces,” on page 32-1](#).

**LOCK RESOURCE TYPE**

The type of resource on which the lock detail request is made. Valid values are shown in [Table 33-1 on page 33-3](#).

**LOCK RESOURCE NAME**

The name on which the lock detail request is made. Each part of the lock resource name is printed on a separate line. The abbreviations shown in the report are explained, in alphabetical order, in [Table 33-2 on page 33-5](#).

**TOTAL REQ**

The total number of lock requests. The sum is calculated by adding the number of request types like LOCK, UNLOCK, CHANGE, or OTHER request types.

**LOCAL**

The number of lock requests that were not sent to cross-system extended services (XES). The sum is calculated by adding the number of request types like LOCK, UNLOCK, CHANGE, or OTHER request types found in the IRLM FUNCTION CODE if data indicates that the request was not sent to z/OS XES.

**XES**

The number of lock requests that were sent to cross-system extended services (XES). The sum is calculated by adding the number of request types like LOCK, UNLOCK, CHANGE, or OTHER request types found in the IRLM FUNCTION CODE if data indicates that the request was sent to z/OS XES.

**REQ TYPE**

The lock request types:

- **LOCK**  Lock function
- **UNLOCK**  Unlock function
- **CHNGE**  Change function
- **OTHER**  Any other functions

**LOCK STATE**

The lock state can be:

- **IS**  Intent share
## Locking Activity - Lock Detail Report

<table>
<thead>
<tr>
<th>LOCK STATE</th>
<th>LOCK DURATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>IX</td>
<td>Intent exclusive</td>
</tr>
<tr>
<td>SIX</td>
<td>Share with intent exclusive</td>
</tr>
<tr>
<td>NSU</td>
<td>Non shared update</td>
</tr>
<tr>
<td>S</td>
<td>Share</td>
</tr>
<tr>
<td>X</td>
<td>Exclusive</td>
</tr>
<tr>
<td>U</td>
<td>Update</td>
</tr>
</tbody>
</table>

### LOCK DURATION

The lock duration can be:

- **CMT** Commit
- **CMT+1** Commit + 1
- **ALLOC** Allocation
- **MAN** Manual
- **MAN+1** Manual + 1
- **OTHER** Other

### COND

The number of lock requests with request type or mode CONDITIONAL.

### AUTREL

The number of lock requests with request type or mode AUTOMATIC RELEASE.

### TOTAL

Total lines are printed regardless of the number of different IDs printed before even if the report presents only a single ID.

### Locking Detail Report

"Locking Detail Report" shows a sample Locking Detail report, produced by the following command:

```
LOCKING REPORT LEVEL (DETAIL)
```

<table>
<thead>
<tr>
<th>DATABASE</th>
<th>TYPE</th>
<th>NAME</th>
<th>LOCK RESOURCE</th>
<th>LOCAL LOCK IS</th>
<th>UNLOCK IS</th>
<th>IX</th>
<th>SIX</th>
<th>NSU</th>
<th>MAN</th>
<th>CMT</th>
<th>ALLOC</th>
<th>COND</th>
<th>AUTREL</th>
</tr>
</thead>
<tbody>
<tr>
<td>'BLANK'</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXCP UPD</td>
<td>RMID=14</td>
<td>32 0 16 16 0 0 0 0 32 0 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>HASH=1</td>
<td>32 0 0 16 16 0 0 0 0 0 0 0 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXCP UPD</td>
<td>RMID=14</td>
<td>32 0 16 16 0 0 0 0 32 0 0 0 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>HASH=2</td>
<td>32 0 0 16 16 0 0 0 0 0 0 0 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXCP UPD</td>
<td>RMID=14</td>
<td>32 0 16 16 0 0 0 0 32 0 0 0 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>HASH=3</td>
<td>32 0 0 16 16 0 0 0 0 0 0 0 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXCP UPD</td>
<td>RMID=14</td>
<td>32 0 16 16 0 0 0 0 32 0 0 0 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>HASH=4</td>
<td>32 0 0 16 16 0 0 0 0 0 0 0 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXCP UPD</td>
<td>RMID=14</td>
<td>32 0 16 16 0 0 0 0 32 0 0 0 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>HASH=5</td>
<td>32 0 0 16 16 0 0 0 0 0 0 0 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXCP UPD</td>
<td>RMID=14</td>
<td>32 0 16 16 0 0 0 0 32 0 0 0 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Locking Activity - Lock Detail Report

<table>
<thead>
<tr>
<th>HASH</th>
<th>Lock Duration</th>
<th>Lock State</th>
<th>TYPE</th>
<th>Local</th>
<th>Other</th>
<th>SIX</th>
<th>NSU</th>
<th>MAN</th>
<th>CMT</th>
<th>ALLOC</th>
<th>COND</th>
<th>OTHER</th>
<th>AUTREL</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

LOCATION: 0MPD0001
LOCATION: OMEGAMON XE for DB2 PERFORMANCE EXPERT (V5R2M0)
GROUP: N/P LOCKING REPORT - DETAIL REQUESTED FROM: NOT SPECIFIED
MEMBER: DB2U001 TO: NOT SPECIFIED
SUBSYSTEM: DB2U ORDER: DATABASE-PAGESET INTERVAL FROM: 03/04/11 08:54:13.83 DB2 VERSION: V9 SCOPE: MEMBER TO: 03/04/11 09:02:47.51
Locking Activity - Lock Detail Report

<table>
<thead>
<tr>
<th>LOCATION: OMPD0001</th>
<th>OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0)</th>
<th>PAGE: 1-4</th>
</tr>
</thead>
<tbody>
<tr>
<td>GROUP: N/P</td>
<td>LOCKING REPORT - DETAIL</td>
<td>REQUESTED FROM: NOT SPECIFIED</td>
</tr>
<tr>
<td>MEMBER: DB2U001</td>
<td>TO: NOT SPECIFIED</td>
<td></td>
</tr>
<tr>
<td>SUBSYSTEM: DB2U</td>
<td>ORDER: DATABASE-PAGESET</td>
<td></td>
</tr>
<tr>
<td>DB2 VERSION: V9</td>
<td>SCOPE: MEMBER</td>
<td></td>
</tr>
<tr>
<td>TO: 03/04/11 09:02:47.51</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Database**

<table>
<thead>
<tr>
<th>Database</th>
<th>Lock Resource</th>
<th>Total</th>
<th>Local</th>
<th>Lock</th>
<th>Unlock IS</th>
<th>IX</th>
<th>SIX</th>
<th>NSU</th>
<th>Man</th>
<th>CMT</th>
<th>Alloc</th>
<th>Cond</th>
</tr>
</thead>
<tbody>
<tr>
<td>GBP S/S</td>
<td>GBP32K</td>
<td>36</td>
<td>0</td>
<td>18</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GBP S/S</td>
<td>GBP8K0</td>
<td>36</td>
<td>0</td>
<td>18</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>N/P</td>
<td>N/P</td>
<td>162916</td>
<td>153K</td>
<td>4705</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SCA</td>
<td>SCA ACCS</td>
<td>56</td>
<td>0</td>
<td>50</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SKPT</td>
<td>SKPT PLAN=DCX1P001</td>
<td>4</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SKPT</td>
<td>SKPT PLAN=K02PLN</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Summary**

<table>
<thead>
<tr>
<th>Total</th>
<th>Blank</th>
<th>361905</th>
<th>286K</th>
<th>168K</th>
<th>189K</th>
<th>145</th>
<th>215</th>
<th>130</th>
<th>73304</th>
<th>262K</th>
<th>0</th>
<th>128K</th>
</tr>
</thead>
</table>

*TOTAL* `BLANK` 361905 286K 168K 189K 145 215 130 0 73304 262K 0 128K

Chapter 33. Locking Activity Report 33-19
### Locking Activity - Lock Detail Report

<table>
<thead>
<tr>
<th>DATABASE PAGESET TYPE NAME</th>
<th>REQ</th>
<th>LOCAL LOCK RESOURCE</th>
<th>TOTAL LOCK DURATION</th>
<th>LOCK STATE</th>
<th>UNLOCK IS</th>
<th>IX</th>
<th>SIX</th>
<th>NSU</th>
<th>MAN</th>
<th>CMT</th>
<th>ALLOC</th>
<th>COND</th>
<th>AUTREL</th>
</tr>
</thead>
<tbody>
<tr>
<td>PAGELCK PART= 1</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>PAGE='X'005FB3'</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PAGELCK PART= 1</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>PAGE='X'005FB8'</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PAGELCK PART= 1</td>
<td>6</td>
<td>0</td>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>PAGE='X'005F51'</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PAGELCK PART= 1</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>PAGE='X'005F92'</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PAGELCK PART= 1</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>PAGE='X'005F94'</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PAGELCK PART= 1</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>PAGE='X'005FC'</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**SUM OF DSNLX02**

<table>
<thead>
<tr>
<th>DATABASE PAGESET TYPE NAME</th>
<th>REQ</th>
<th>LOCAL LOCK RESOURCE</th>
<th>TOTAL LOCK DURATION</th>
<th>LOCK STATE</th>
<th>UNLOCK IS</th>
<th>IX</th>
<th>SIX</th>
<th>NSU</th>
<th>MAN</th>
<th>CMT</th>
<th>ALLOC</th>
<th>COND</th>
<th>AUTREL</th>
</tr>
</thead>
<tbody>
<tr>
<td>PAGELCK PART= 1</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>PAGE='X'0096B0'</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DATAPAGE PAGE='X'0096B0'</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DATAPAGE PAGE='X'009BB3'</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DATAPAGE PAGE='X'00995A'</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DATAPAGE PAGE='X'009957'</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DATAPAGE PAGE='X'009958'</td>
<td>9</td>
<td>0</td>
<td>9</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DATAPAGE PAGE='X'009959'</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DATAPAGE PAGE='X'001D6'</td>
<td>3</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DATAPAGE PAGE='X'001D9'</td>
<td>3</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DATAPAGE PAGE='X'007F6D'</td>
<td>24</td>
<td>0</td>
<td>14</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DATAPAGE PAGE='X'007F66'</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DATAPAGE PAGE='X'007F6F'</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DATAPAGE PAGE='X'007F6'</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DATAPAGE PAGE='X'007F70'</td>
<td>16</td>
<td>0</td>
<td>11</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**SUM OF DSNLX02**

LOCATION: OMPD0001
GROUP: N/P
MEMBER: DB2U001
SUBSYSTEM: DB2U
ORDER: DATABASE-PAGESET
SCOPE: MEMBER
DB2 VERSION: V9

LOCATION: OMPD0001
GROUP: N/P
MEMBER: DB2U001
SUBSYSTEM: DB2U
ORDER: DATABASE-PAGESET
SCOPE: MEMBER
DB2 VERSION: V9

LOCATION: OMPD0001
GROUP: N/P
MEMBER: DB2U001
SUBSYSTEM: DB2U
ORDER: DATABASE-PAGESET
SCOPE: MEMBER
DB2 VERSION: V9

LOCATION: OMPD0001
GROUP: N/P
MEMBER: DB2U001
SUBSYSTEM: DB2U
ORDER: DATABASE-PAGESET
SCOPE: MEMBER
DB2 VERSION: V9
## Locking Activity - Lock Detail Report

### Database Pageset

<table>
<thead>
<tr>
<th>Database Page</th>
<th>Type</th>
<th>Name</th>
<th>Req</th>
<th>Xes</th>
<th>Change</th>
<th>Unlock</th>
<th>IS</th>
<th>IX</th>
<th>SIX</th>
<th>NSU</th>
<th>MAN</th>
<th>CMT+1</th>
<th>Alloc</th>
<th>Cond</th>
<th><strong>-Lock Duration-</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>PAGE=PAGESET</td>
<td>N/P</td>
<td>0000</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

**TOTAL**

- **SUM OF SYSLGRNX**: 198 6 160 6 0 59 0 0 38 154 0 98
- **SUM OF DSNACH01**: 4 2 2 2 0 400 1 200 0

### Location

- **LOCATION**: OMPD0001 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0)
- **GROUP**: N/P LOCKING REPORT - DETAIL
- **MEMBER**: DB2U001 TO: NOT SPECIFIED
- **SUBSYSTEM**: DB2U ORDER: DATABASE-PAGESET INTERVAL FROM: 03/04/11 08:54:13.83 DB2 VERSION: V9 SCOPE: MEMBER TO: 03/04/11 09:02:47.51

### Page Locking Activity

- **TOTAL**
- **GRAND TOTAL**: 2160727 1050K 1576K 568K 244K 70460 130 0 537K 891K 37113 566K 1111K 15686 781 983K 395K 5621 62188 5017 629K 40093

LOCKING REPORT COMPLETE
Chapter 34. Locking Trace

Note: For an introduction to the Locking report set and general locking information refer to the Reporting User's Guide. It also provides information on input to locking.

The layout for locking traces is the same for each trace apart from the event-specific data. “Layout of a Deadlock Trace” shows the general layout of the locking trace and describes the common fields.

Layout of a Deadlock Trace

Field description

Here you find a description of all fields except for the trace header and the OMEGAMON XE for DB2 PE identifiers, for details of the trace header see Chapter 32, “The Locking Header of Reports and Traces,” on page 32-1. The descriptions start with the timestamp block and move to the right.

reporttype

This can be:

- DEADLOCK
- TIMEOUT
- LOCKOUT
- SUSPENSION
- DETAIL

EVENT TIMESTAMP

The time at which the event occurred. The trace is sorted and printed in the order of this timestamp. The format of this timestamp is hh:mm:ss.nnnnnn.

RELATED TIMESTAMP

The timestamp of the suspended request that was selected as the victim of this event. This field only shows a value for suspension and detail traces. For other traces, this field always shows N/P because the related suspension event is not reported.

EVENT

The Locking event. This varies according to the type of trace.
LOCK RESOURCE TYPE
The type of locked resource. The values for the locked resource types are shown in Table 33-1 on page 33-3.

LOCK RESOURCE NAME
The name of the resource. Each part of the lock resource name is printed on a separate line. The format of the name depends on the resource type as shown in Table 33-2 on page 33-5.

EVENT SPECIFIC DATA
The layout and content of the event specific data varies according to the reported event and is described in the sections following.

“Deadlock Trace” on page 34-3
The deadlock trace contains an entry for every occurrence of a deadlock during a specified time period.

“Timeout Trace” on page 34-11
The timeout trace shows when a timeout occurred and provides details of the resource involved in the timeout event and information about the threads that held the resource or waited to use the resource.

“Lockout Trace” on page 34-16
The lockout trace contains details of timeout and deadlock events.

“Lock Suspension Trace” on page 34-17
The lock suspension trace identifies applications that have been suspended after a lock was requested on a resource that is not available.

“Lock Detail Trace Data” on page 34-30
The lock detail trace describes all locking events in a DB2 system. It includes those that can be viewed in suspension, timeout, or deadlock traces. This trace gives you a global view of the entire locking activity in the system.
Deadlock Trace

The deadlock trace contains an entry for every occurrence of a deadlock during a specified time period.

The trace shows when the deadlock occurred and provides details on the resources involved in the deadlock and information about the threads that held the resource or waited to use the resource. If the resource was held by more than one agent and not all of them were actively involved in the deadlock, the holder data cannot be determined and is not printed.

The data specific to the deadlock. For each resource involved in a deadlock there is a block of waiter's data and a block of blocker's data.

A blocker is a thread that prevents the victim getting its lock. The blocker can be a holder of the lock or another waiter (one that came in before the victim) that is incompatible with the holder's lock.

There is no correlation between the number of deadlock events reported by Locking reports and traces and the number of deadlocks reported in Accounting and Statistics reports. Whereas Accounting and Statistics reports count all deadlock occurrences, regardless of how they resolve, Locking reports only those deadlocks that were resolved by DB2. DB2 can resolve a deadlock either by making a process roll back, thereby releasing the locks it holds on resources, or by requesting a process to terminate.

The format of the deadlock-specific data is shown in “The Format of Deadlock-Specific Data” on page 34-4.
Trace Data Specific to Deadlock Event

This topic describes the trace data that is specific to the Deadlock Event.

The Format of Deadlock-Specific Data

The following example shows the layout of Deadlock-Specific data.

COUNTER =XXXXX WAITERS =XXXXX
TSTAMP =MM/DD/YY HH:MM:SS.ss
HASH =X'HHHHHHHH'
--------- BLOCKER is HOLDER -----------
LUW=XXXXXXXX.XXXXXXXX.XXXXXXXXXXXX
MEMBER =XXXXXXXXX CONNECT =XXXXXXXX
PLANNAME=XXXXXXXXX CORRID =XXXXXXXXXXX
DURATION=XXXXXXXXX PRIMAUTH=XXXXXXXX
STATE =XXXXX STMTINFO=XXXXXXXX
ENDUSER =XXXXXXXXXXXX
WSNAME =XXXXXXXXXXXXX
TRANSAC=XXXXXXXXXXXXXXXXXXXXXXXXX
LOGNAME=xxxxxxx
COLLID =xxxxxxxxxxxxxxxxx
LOCATION=xxxxxxxxxxxxxxxx
CONTOKEN=X'xxxxxxxxxxxxxxxx'
STMTID =X'xxxxxxxxxxxxxxxx'

--------- WAITER  -----------
LUW=XXXXXXXX.XXXXXXXX.XXXXXXXXXXXX
MEMBER =XXXXXXXXX CONNECT =XXXXXXXX
PLANNAME=XXXXXXXXX CORRID =XXXXXXXXXXX
DURATION=XXXXXXXXX PRIMAUTH=XXXXXXXX
REQUEST =XXXXXX WORTH =XXXXX
STATE =XXXXX STMTINFO=XXXXXXXX
ENDUSER =XXXXXXXXXXXX
WSNAME =XXXXXXXXXXXXX
TRANSAC=XXXXXXXXXXXXXXXXXXXXXXXXX
LOGNAME=xxxxxxx
COLLID =xxxxxxxxxxxxxxxxx
LOCATION=xxxxxxxxxxxxxxxx
CONTOKEN=X'xxxxxxxxxxxxxxxx'
STMTID =X'xxxxxxxxxxxxxxxx'

Field description

The individual fields have the following meaning:

COUNTER
   The deadlock interval counter.

WAITERS
   The number of waiters involved in the deadlock.

TSTAMP
   The time when the deadlock occurred.

HASH
   The lock resource hash value.

LUW
   The ID of the blocker's or waiter's logical unit of work.

MEMBER
   The blocker's or waiter's member name. In a non-data-sharing environment, this field contains N/P.

CONNECT
   The holder's or waiter's connection name.

PLANNAME
   The blocker's or waiter's plan name.

CORRID
   The blocker's or waiter's correlation name.
DURATION

The lock duration of the deadlock blocker or waiter. Valid values are shown in Table 34-1.

Table 34-1. Lock Duration

<table>
<thead>
<tr>
<th>Duration</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MANUAL</td>
<td>Varies depending on the ISOLATION parameter</td>
</tr>
<tr>
<td>MANUAL+1</td>
<td>Temporary change of consistency level from CS to RR during bind and DDL</td>
</tr>
<tr>
<td>COMMIT</td>
<td>Until commit</td>
</tr>
<tr>
<td>COMMIT+1</td>
<td>Past commit; applies to locks needed to maintain the position for a cursor opened WITH HOLD</td>
</tr>
<tr>
<td>ALLOCATN</td>
<td>Until deallocation</td>
</tr>
<tr>
<td>PLAN</td>
<td>For the duration of the plan</td>
</tr>
<tr>
<td>UTILITY</td>
<td>For the duration of the utility execution</td>
</tr>
<tr>
<td>INTEREST</td>
<td>For the duration of P-Locks</td>
</tr>
<tr>
<td>FREE ALL</td>
<td>Until all locks are freed</td>
</tr>
</tbody>
</table>

PRIMAUTH

The primary authorization ID of the thread.

REQUEST

The waiter's request, which can be one of the following:
- LOCK
- UNLOCK
- CHANGE

WORTH

The waiter's worth value assigned by DB2.

STATE

The holder's or waiter's state or mode of the lock applied to the resource. Valid values are shown in Table 34-2.

Table 34-2. Lock State

<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UPS</td>
<td>Unprotected share</td>
</tr>
<tr>
<td>IS</td>
<td>Intent share</td>
</tr>
<tr>
<td>IX</td>
<td>Intent exclusive</td>
</tr>
<tr>
<td>S</td>
<td>Share</td>
</tr>
<tr>
<td>U</td>
<td>Update</td>
</tr>
<tr>
<td>SIX</td>
<td>Share with intent exclusive</td>
</tr>
<tr>
<td>NSU</td>
<td>Nonshared update</td>
</tr>
<tr>
<td>X</td>
<td>Exclusive</td>
</tr>
</tbody>
</table>

STMTINFO

The statement information of the holder or waiter.

ENDUSER

End user's user ID. This field is not shown when this information is not present.
**Locking Activity - Trace**

**WSNAME**
End user's workstation name. This field is not shown when this information is not present.

**TRANSAC**
The end user's transaction name. This field is not shown when this information is not present.

**PROGNAME**
The program name can be one of the following:
- The name of the blocker's program that is currently in control at the time of the deadlock and not necessarily the program that acquired the lock.
- The waiter's program that is contending the resource.

**COLLID**
The collection identifier can be one of the following:
- The package collection ID of the blocker's program that is currently in control at the time of the deadlock and not necessarily the program that acquired the lock.
- The package collection ID of the waiter's program that is contending the resource.

**LOCATION**
The location can be one of the following:
- The location of the blocker's program that is currently in control at the time of the deadlock and not necessarily the program that acquired the lock.
- The location of the waiter's program that is contending the resource.

**CONTOKEN**
The consistency token can be one of the following:
- The consistency token of the blocker's program that is currently in control at the time of the deadlock and not necessarily the program that acquired the lock.
- The consistency token of the waiter's program that is contending the resource.

**STMTID**
The statement ID of the holder or waiter.
Deadlock Trace Example

"Deadlock trace example" shows a sample deadlock trace, produced by the following command:

```
LOCKING TRACE LEVEL (DEADLOCK)
```

Deadlock trace example

This is a sample deadlock trace:

```
LOCATION: OMPDA21 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-1
GROUP: DBDAGROU LOCKING TRACE - DEADLOCK REQUESTED FROM: NOT SPECIFIED
MEMBER: DA31MEMB TO: NOT SPECIFIED
SUBSYSTEM: DA31 ACTUAL FROM: 11/03/09 17:30:58.10
DB2 VERSION: V9 SCOPE: MEMBER PAGE DATE: 11/03/09

PRIMAUTH CORRNMBR INSTANCE EVENT TIMESTAMP --- LOCK RESOURCE --- EVENT SPECIFIC DATA
--- PRIMAUTH CORRNMBR INSTANCE EVENT --- TIMESTAMP EVENT TYPE NAME ---
------------------------------ ----------------- -------- --------- ----------------------- ----------------------------------------
SHA 'java' DRDA 17:30:58.10472181 DEADLOCK COUNTER = 5243 WAITERS = 2
SHA 'BLANK' CS06B221EA18 N/P TSTAMP = 11/03/09 17:30:58.10
DISTSERV SERVER TABLE DB = TDKDB HASH = x'00001001' BLOCKER is HOLDER ---VICTIM---
REQLOC ::ffff:9.162.122 OB = 11 REQUEST = CHANGE WORTH = 18 STATE = X STMTINFO=N/A
ENDUSER : sha
WSNAME : miller
TRANSACT: java

DURATION=COMMIT PRIMAUTH=SHA
STATE = S STMTINFO=N/A
ENDUSER = sha
WSNAME = miller
TRANSACT = java
PROGNAME = SYSXAS01
COLLID = NULLID
LOCATION=N/P
CONTOKEN=X'5359534C564C3031'
STMTID = N/A

----------- WAITER ---------------
LUW=G9987A4A.B601.C508B2220AB3 MEMBER = DA31MEMB CONNECT =SERVER
PLANNAME=DISTSERV CORRID = java
DURATION=COMMIT PRIMAUTH=SHA
STATE = X STMTINFO=N/A
ENDUSER = sha
WSNAME = miller
TRANSACT = java
PROGNAME = SYSXAS01
COLLID = NULLID
LOCATION=N/P
CONTOKEN=X'5359534C564C3031'
STMTID = N/A

----------- WAITER ---------------
LUW=G9987A4A.B601.C508B2220AB3 MEMBER = DA31MEMB CONNECT =SERVER
PLANNAME=DISTSERV CORRID = java
DURATION=COMMIT PRIMAUTH=SHA
STATE = X STMTINFO=N/A
ENDUSER = sha
WSNAME = miller
TRANSACT = java
PROGNAME = SYSXAS01
COLLID = NULLID
LOCATION=N/P
CONTOKEN=X'5359534C564C3031'
STMTID = N/A

----------- WAITER ---------------
LUW=G9987A4A.B601.C508B2220AB3 MEMBER = DA31MEMB CONNECT =SERVER
PLANNAME=DISTSERV CORRID = java
DURATION=COMMIT PRIMAUTH=SHA
STATE = X STMTINFO=N/A
ENDUSER = sha
WSNAME = miller
TRANSACT = java
PROGNAME = SYSXAS01
COLLID = NULLID
LOCATION=N/P
CONTOKEN=X'5359534C564C3031'
STMTID = N/A
```

Chapter 34. Locking Trace 34-7
Locking Activity - Trace

SHA java ORDA 17:31:08.12271550 DEADLOCK
SHA 'BLANK' C508B2220AB3 N/P
DISTSERV SERVER TABLE DB =TDKDB
REQLOC ::FFFF:9.152.122 DB =11
ENDUSER :sha
WSNAME :miller
TRANSACT:java

COUNTER = 5245 WAITERS = 2
TSTAMP =11/03/09 17:31:08.10
HASH ="x'00000000'"

----------- BLOCKER is HOLDER ------>VICTIM-----
LUN=69874AA.AB61.C508B2220AB3
MEMBER =DA31MGB CONNECT +SERVER
PLANNAME=DISTSERV CORRID =java
DURATION=COMMIT PRIMAUTH=SHA
STATE =S STMTINFO=N/A
ENDUSER =sha
WSNAME =miller
TRANSACT:java
PROGNAME=SYSKA501
COLLID =NULLID
LOCATION=N/P
CONTOKEN=X'5359534C564C3031'
STMTID =N/A

----------- WAITER ----------------
LUN=69874AA.AB61.C508B2220AB3
MEMBER =DA31MGB CONNECT +SERVER
PLANNAME=DISTSERV CORRID =java
DURATION=COMMIT PRIMAUTH=SHA
REQUEST =CHANGE WORTH = 18
STATE =X STMTINFO=N/A
ENDUSER =sha
WSNAME =miller
TRANSACT:java
PROGNAME=SYSKA501
COLLID =NULLID
LOCATION=N/P
CONTOKEN=X'5359534C564C3031'
STMTID =N/A

----------- WAITER ----------------
LUN=69874AA.AB61.C508B2220AB3
MEMBER =DA31MGB CONNECT +SERVER
PLANNAME=DISTSERV CORRID =java
DURATION=COMMIT PRIMAUTH=SHA
REQUEST =CHANGE WORTH = 17
STATE =X STMTINFO=N/A
ENDUSER =sha
WSNAME =miller
TRANSACT:java
PROGNAME=SYSKA501
COLLID =NULLID
LOCATION=N/P
CONTOKEN=X'5359534C564C3031'
STMTID =N/A

SHA java ORDA 17:31:13.10897581 DEADLOCK
SHA 'BLANK' C50864AAC7BA N/P
DISTSERV SERVER TABLE DB =TDKDB
REQLOC ::FFFF:9.152.122 DB =11
ENDUSER :sha
WSNAME :miller
TRANSACT:java

COUNTER = 5246 WAITERS = 2
TSTAMP =11/03/09 17:31:13.10
HASH ="x'00000000'"

----------- BLOCKER is HOLDER ------>VICTIM-----
LUN=69874AA.AB61.C508B2220AB3
MEMBER =DA31MGB CONNECT +SERVER
PLANNAME=DISTSERV CORRID =java
DURATION=COMMIT PRIMAUTH=SHA
STATE =S STMTINFO=N/A
ENDUSER =sha
WSNAME =miller
TRANSACT:java
PROGNAME=SYSKA501
COLLID =NULLID
LOCATION=N/P
CONTOKEN=X'5359534C564C3031'
STMTID =N/A

----------- WAITER ----------------
LUN=69874AA.AB61.C508B2220AB3
MEMBER =DA31MGB CONNECT +SERVER
PLANNAME=DISTSERV CORRID =java
DURATION=COMMIT PRIMAUTH=SHA
REQUEST =CHANGE WORTH = 18
STATE =X STMTINFO=N/A
ENDUSER =sha
WSNAME =miller
TRANSACT:java
PROGNAME=SYSKA501
COLLID =NULLID
LOCATION=N/P
CONTOKEN=X'5359534C564C3031'
STMTID =N/A

----------- WAITER ----------------
LUN=69874AA.AB61.C508B2220AB3
MEMBER =DA31MGB CONNECT +SERVER
PLANNAME=DISTSERV CORRID =java
DURATION=COMMIT PRIMAUTH=SHA
REQUEST =CHANGE WORTH = 17
STATE =X STMTINFO=N/A

34-8 OMEGAMON XE for DB2 PE & PM: Report Reference
Locking Activity - Trace

Enduser: sha
WSName: miller
TRANSAC: java
PROGNAME=SYSKA501
COLLID=NULLID
LOCATION=N/P
CONTOKEN=X'5359534C564C3031'
STMTID = N/A

<table>
<thead>
<tr>
<th>TABLE</th>
<th>DB</th>
<th>RSS</th>
<th>OB</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB</td>
<td>TDKDB</td>
<td></td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>DB</td>
<td>TDKDB</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Hash = X'00010081'
----------- BLOCKER IS WAITER -----------
LUW=G99874B.AB97_505B2DE670E
MEMBER = DA3MEMB CONNECT = SERVER
PLANNAME=DISTSERV CORRID = java
DURATION=COMMIT PRIMAUTH=SHA
STATE = X STMTINFO=N/A

Enduser: sha
WSName: miller
TRANSAC: java
PROGNAME=SYSKA501
COLLID=NULLID
LOCATION=N/P
CONTOKEN=X'5359534C564C3031'
STMTID = N/A

REQUEST = CHANGE WORTH = 17
STATE = X STMTINFO=N/A

Enduser: sha
WSName: miller
TRANSAC: java
PROGNAME=SYSKA501
COLLID=NULLID
LOCATION=N/P
CONTOKEN=X'5359534C564C3031'
STMTID = N/A

Member = DA3MEMB CONNECT = SERVER
PLANNAME=DISTSERV CORRID = java
DURATION=COMMIT PRIMAUTH=SHA
STATE = X STMTINFO=N/A

Enduser: sha
WSName: miller
TRANSAC: java
PROGNAME=SYSKA501
COLLID=NULLID
LOCATION=N/P
CONTOKEN=X'5359534C564C3031'
STMTID = N/A

Member = DA3MEMB CONNECT = SERVER
PLANNAME=DISTSERV CORRID = java
DURATION=COMMIT PRIMAUTH=SHA
STATE = X STMTINFO=N/A

Enduser: sha
WSName: miller
TRANSAC: java
PROGNAME=SYSKA501
COLLID=NULLID
LOCATION=N/P
CONTOKEN=X'5359534C564C3031'
STMTID = N/A

Member = DA3MEMB CONNECT = SERVER
PLANNAME=DISTSERV CORRID = java
DURATION=COMMIT PRIMAUTH=SHA
STATE = X STMTINFO=N/A

Enduser: sha
WSName: miller
TRANSAC: java
PROGNAME=SYSKA501
COLLID=NULLID
LOCATION=N/P
CONTOKEN=X'5359534C564C3031'
STMTID = N/A

Member = DA3MEMB CONNECT = SERVER
PLANNAME=DISTSERV CORRID = java
DURATION=COMMIT PRIMAUTH=SHA
STATE = X STMTINFO=N/A

Enduser: sha
WSName: miller
TRANSAC: java
PROGNAME=SYSKA501
COLLID=NULLID
LOCATION=N/P
CONTOKEN=X'5359534C564C3031'
STMTID = N/A

Member = DA3MEMB CONNECT = SERVER
PLANNAME=DISTSERV CORRID = java
DURATION=COMMIT PRIMAUTH=SHA
STATE = X STMTINFO=N/A

Enduser: sha
WSName: miller
TRANSAC: java
PROGNAME=SYSKA501
COLLID=NULLID
LOCATION=N/P
CONTOKEN=X'5359534C564C3031'
STMTID = N/A
Locking Activity - Trace

مادة 관한 - Trace

-ruqis = +victim-
lu9 = 698844, 85, 4808211818
memex = da31mmb, conek = server
planena = distserv, corid = java
duration = commit, primauth = sha
request = change, worth = 17
state = x
stmtinfo = n/a
enduser = sha
wsname = miller
transac = java
prgname = syska501
collid = nullid
location = n/p
contoken = 1's359534c56ac3031'
stmtid = n/a

Locking Trace Complete
Timeout Trace

The timeout trace shows when a timeout occurred and provides details of the resource involved in the timeout event and information about the threads that held the resource or waited to use the resource.

The following sections show the layout of event-specific information for a timeout trace and describe the fields reported. At the end of this topic you find an example of a timeout trace.

"Trace Data Specific to Timeout Event" on page 34-12
"Timeout Trace Example" on page 34-15
Locking Activity - Trace

Trace Data Specific to Timeout Event

The details related to the timeout. The format of the timeout-specific data is shown in "Format of timeout-specific data."

Format of timeout-specific data

The following example shows details related to the timeout.

```
REQUEST =LOCK UNCONDITIONAL
STATE =IS ZPARM INTERVAL = 30
DURATION = COMMIT INTERV.COUNTER = 1
HASH = x’00015F0F’
STMTINFO = DYNAMIC
STMTID = x’000000000000A341’
------------- HOLDERS/WAITERS -------------
HOLDER
LUW=DEIBMIPS.IPUAX32.C6215376BB44
MEMBER =SZ32 CONNECT =BATCH
PLANNAME=DSNTIA10 CORRID =YULT3978
DURATION = COMMIT PRIMAUTH=SKA
STATE =X STMTINFO = DYNAMIC
STMTID = x’000000000000A312’
```

Field description

The individual fields have the following meaning:

REQUEST

The timeout request, consists of one of the following:

- LOCK
- CHANGE

Followed by the timeout attribute CONDITIONAL or UNCONDITIONAL.

STATE

The state or mode of the lock applied to the resource. Valid values are shown in Table 34-2 on page 34-5.

ZPARM INTERVAL

The timeout interval (ZPARM value), which is the timeout value specified on the installation panel DSNTIPX or in the ZPARM name STORTIME in DSN6SYSP.

DURATION

The length of time for which the lock was held. Valid values are shown in Table 34-3.

Table 34-3. Lock Duration

<table>
<thead>
<tr>
<th>Duration</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MANUAL</td>
<td>Varies depending on the ISOLATION parameter</td>
</tr>
<tr>
<td>MANUAL+1</td>
<td>Temporary change of consistency level from CS to RR during bind and DDL</td>
</tr>
<tr>
<td>COMMIT</td>
<td>Until commit</td>
</tr>
<tr>
<td>COMMIT+1</td>
<td>Past commit; applies to locks needed to maintain the position for a cursor opened WITH HOLD</td>
</tr>
<tr>
<td>ALLOCATN</td>
<td>Until deallocation</td>
</tr>
<tr>
<td>PLAN</td>
<td>Lock held for the duration of the plan</td>
</tr>
<tr>
<td>UTILITY</td>
<td>For the duration of the utility execution</td>
</tr>
<tr>
<td>FREE ALL</td>
<td>Until all locks are freed</td>
</tr>
<tr>
<td>X’00’</td>
<td>The suspension reason is a retained lock</td>
</tr>
</tbody>
</table>
The DURATION attribute controls when locks are released. As a general rule, a lock is only released when an agent makes an unlock request with a duration longer than, or equal to, the longest lock duration specified for the resource by that agent.

You increase lock durations using either a lock request or a change request. Lock durations are decreased using a change request.

**INTERV.COUNTER**
The number of timeout intervals that can occur before the agent is timed out.

**HASH**
The lock resource hash value.

**STMTINFO**
The waiter's statement information. Possible values are:

- **STATIC**
The statement is of type static.
- **DYNAMIC**
The statement is of type dynamic.

**STMTID**
The cached statement ID for the statement waiting for the resource. A value of zero indicates that the client did not supply this information.

**Fields that are printed for each holder/waiter**

The following fields are printed for each **holder/waiter** of the reported lock resource:

**LUW**
The ID of the holder's or waiter's logical unit of work. If the reason for the suspension is a retained lock, this field contains the word SYSTEM.

**MEMBER**
The holder's or waiter's DB2 member name. In a non-data-sharing environment, N/P is printed.

**CONNECT**
The holder's or waiter's connection name. If the reason for the suspension is a retained lock, this field contains the word SYSTEM.

**PLANNNAME**
The holder's or waiter's plan name. If the reason for the suspension is a retained lock, this field contains the word SYSTEM.

**CORRID**
The holder's or waiter's correlation identifier. If the reason for the suspension is a retained lock, this field contains the word SYSTEM.

**DURATION**
The lock duration of the timeout holder or waiter. Valid values are shown in Table 34-3 on page 34-12.

**PRIMAUTH**
The primary authorization ID.

**STATE**
The holder's or waiter's state or mode of the lock applied to the resource. Valid values are shown in Table 34-2 on page 34-5.
Locking Activity - Trace

**STMTINFO**
The holder’s statement information. Possible values are:

**STATIC**
The statement is of type static.

**DYNAMIC**
The statement is of type dynamic.

**STMTID**
The cached statement ID for the statement holding the resource. A value of zero indicates that the client did not supply this information.
Timeout Trace Example

"Timeout trace example" shows a sample Timeout trace, produced by the following command:

```
LOCKING TRACE LEVEL (TIMEOUT)
```

Timeout trace example

This is a sample Timeout trace:

```
LOCATION: STLEC1 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-1
GROUP: N/P LOCKING TRACE - TIMEOUT REQUESTED FROM: NOT SPECIFIED TO: NOT SPECIFIED
MEMBER: N/P TO: NOT SPECIFIED
SUBSYSTEM: VA1A ACTUAL FROM: 06/04/09 00:15:44.20
DB2 VERSION: V10 SCOPE: MEMBER PAGE DATE: 06/04/09

PRIMAUTH CORRNAME CONNTYPE
ORIGAUTH CONNMEMBER INSTANCE EVENT TIMESTAMP -- -LOCK RESOURCE--
PLANNAME CONNECT RELATED TIMESTAMP EVENT TYPE NAME EVENT SPECIFIC DATA
------------------------------ ----------------- -------- --------- ----------------------- ----------------------------------------
SYSADM L829UTT2 TSO 00:15:44.20054922 TIMEOUT SKPT REQUEST =LOCK UNCONDITIONAL
SYSADM 'BLANK' C448AF2328DD N/P STATE =U ZPARM INTERVAL= 60
DSNTEP3 BATCH CTKN=0000000000000000 DURATION=MANUAL INTERV.COUNTER= 1
------------ HOLDERS/WAITERS ------------
HOLDER
LUW=USIBMSY.SYEC1DB2.C448AF1E53C6
MEMBER =N/P CONNECT =BATCH
PLANNAME=DSNTEP3 CORRID =L829UTT1 DURATION=COMMIT PRIMAUTH=SYSADM
STATE =X STMTINFO=DYNAMIC STMTID =X'0000000000000001'

LOCKING TRACE COMPLETE
```
Lockout Trace

The lockout trace contains details of timeout and deadlock events.

You generate it by using the following command:

```
... LOCKING TRACE LEVEL (LOCKOUT) ...
```

For information on the layout of a lockout trace, refer to "Deadlock Trace" on page 34-3 and "Timeout Trace" on page 34-11.

### Lockout Trace

Here is an example of a lockout trace.

```plaintext
LOCATION: STLEC1 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (VSAM) PAGE: 1-1
GROUP: N/P LOCKING TRACE - LOCKOUT LEVEL: V10
MEMBER: N/P TO: NOT SPECIFIED
SUBSYSTEM: VA1A ACTUAL FROM: 06/04/09 00:15:44.20
DB2 VERSION: V10 SCOPE: MEMBER PAGE DATE: 06/04/09
PRIMAUTH CORRNAME CONNTYPE ORIGAUTH CORRNMBR INSTANCE
------------------------------ ----------- -------- ------- ----------------------- ----------------------------------------
PLANNAME CONNECT RELATED TIMESTAMP EVENT NAME EVENT SPECIFIC DATA
------------------------------ ----------------- -------- --------- ----------------------- ----------------------------------------
SYSADM L829UTT2 TSO 00:15:44.20054922 TIMEOUT SKPT REQUEST =LOCK UNCONDITIONAL
SYSADM 'BLANK' C448AF2328DD N/P STATE =U ZPARM INTERVAL=60
DSNTEP3 BATCH CTKN=0000000000000000 DURATION=MANUAL INTERV.COUNTER=1
-------- HOLDERS/WAITERS --------
HOLDER LUU=USIMPSY.SYEC1DDB2.C448AF1ES30K MEMBER =N/P CONNECT =BATCH
PLANNAME=DSNTEP3 CORRID =L829UTT1 DURATION=COMMIT PRIMAUTH=SYSADM
STATE =X STMTINFO=DYNAMIC
STMTID =X'0000000000000001'
```

LOCKING TRACE COMPLETE
Lock Suspension Trace

The lock suspension trace identifies applications that have been suspended after a lock was requested on a resource that is not available.

The trace shows an entry for the suspension of each of the following:

- An IRLM request (except when the resource type is a drain lock).
- An IRLM request where the resource type is a drain lock.
- A drain request where the claim count is not zero.
  This suspension occurs when the agent making the drain request has to wait for the claim count on the particular resource to become zero.
- A page latch request.
  This suspension occurs when the agent making the page latch request has to wait for a page that is currently being held by another agent.

The lock suspension trace is produced if level SUSPENSION is specified in the TRACE subcommand and if there is at least one pair of IFCIDs 44/45, 213/214, 215/216, or 226/227 in the input data set satisfying the FROM and TO, and INCLUDE or EXCLUDE criteria.

The following sections show the layout of a lock suspension trace and describe the various fields of the trace. At the end of this topic you find an example of a lock suspension trace.

"Lock Suspension Trace Example" on page 34-18
This topic shows an example of a Lock Suspension trace.

"Lock Suspension Events - Lock, Unlock, Change, and Notify Suspend" on page 34-19
This topic shows the format of data specific to Lock, Unlock, Change, and Notify Suspend for Lock Suspension events. It also describes the fields provided for this event.

"Lock Suspension Events - Lock, Unlock, Change, and Notify Resume" on page 34-21
The format of the data for these events depends on whether these events occurred in a data sharing or non-data-sharing environment.

"Lock Suspension Events - Query Suspend" on page 34-24
This topic shows the format of data specific to Query Suspend for Lock Suspension events. It also describes the fields provided for this event.

"Lock Suspension Events - Query Resume" on page 34-25
This topic shows the format of data specific to Query Resume for Lock Suspension events. It also describes the fields provided for this event.

"Lock Suspension Events - Drain Suspend" on page 34-26
This topic shows the format of data specific to Drain Suspend for Lock Suspension events. It also describes the fields provided for this event.

"Lock Suspension Events - Drain Resume" on page 34-27
This topic shows the format of data specific to Drain Resume for Lock Suspension events. It also describes the fields provided for this event.

"Lock Suspension Events - Latch Suspend" on page 34-28
This topic shows the format of data specific to Latch Suspend for Lock Suspension events. It also describes the fields provided for this event.

"Lock Suspension Events - Latch Resume" on page 34-29
This topic shows the format of data specific to Latch Resume for Lock Suspension events. It also describes the fields provided for this event.
Lock Suspension Trace Example

This topic shows an example of a Lock Suspension trace.

Enter the following command to produce a Lock Suspension trace:

```plaintext
: LOCKING
   TRACE
   LEVEL (SUSPENSION)
```

Lock Suspension trace example

This is a sample layout of a Lock Suspension trace.

```
LOCATION: OMPDA21 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-1
GROUP: N/P LOCKING TRACE - SUSPENSION REQUESTED FROM: NOT SPECIFIED
MEMBER: N/P TO: NOT SPECIFIED
SUBSYSTEM: DA21 ACTUAL FROM: 02/14/11 19:46:19.74
DB2 VERSION: V10 SCOPE: MEMBER PAGE DATE: 02/14/11
PRIMAUTH CORRNAME CONNTYPE
ORIGAUTH CORRNAME INSTANCE
PLANNAME CONNECT RELATED TIMESTAMP EVENT TYPE NAME EVENT SPECIFIC DATA
---------------------------------------------------------------------------
SKA java DRDA 19:46:19.74178259 LOCK SKPT DURATION=COMMIT STATE=S
SKA 'BLANK' 110214171010 SUSPEND ORIG.RSN=LOCAL CONTENTION
DISTSERV SERVER CTKN=5359534C564C3031 HASH =X'00000000'
REGLOC :::FFFF:9.152.122 COLL(HEX)= X'112233445566778899001122334455667788'
ENDUSER :OMPE Testuser PKID(HEX)= X'1122334455667788'
WSNAME :monroe
TRANSACT:BPMWorkload

LOCKING TRACE COMPLETE
```
Lock Suspension Events - Lock, Unlock, Change, and Notify Suspend

This topic shows the format of data specific to Lock, Unlock, Change, and Notify Suspend for Lock Suspension events. It also describes the fields provided for this event.

Format of data specific to Lock, Unlock, Change, and Notify Suspend

This is the sample format for data specific to Lock, Unlock, Change, and Notify Suspend.

```
DURATION=xxxxxxxx STATE=xxxxx XES PROP=x
ORIG.RSN=xxxxx xxxxxxxxxx XES FORC=x
PARENT =xxxxxxxx
HASH =X’hhhhhhhh’
```

Field description

**DURATION**

The length of time the lock is held. Valid values are shown in Table 34-4.

Table 34-4. Lock Duration - IRLM SUSPEND

<table>
<thead>
<tr>
<th>Duration</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>INTEREST</td>
<td>Duration of P-Locks</td>
</tr>
<tr>
<td>MANUAL</td>
<td>Varies depending on the ISOLATION parameter</td>
</tr>
<tr>
<td>MANUAL+1</td>
<td>Temporary change of consistency level from CS to RR during bind and DDL</td>
</tr>
<tr>
<td>COMMIT</td>
<td>Until commit</td>
</tr>
<tr>
<td>COMMIT+1</td>
<td>Past commit; applies to locks needed to maintain the position for a cursor opened WITH HOLD</td>
</tr>
<tr>
<td>ALLOCATN</td>
<td>Until deallocation</td>
</tr>
<tr>
<td>PLAN</td>
<td>For the duration of the plan</td>
</tr>
<tr>
<td>UTILITY</td>
<td>For the duration of the utility execution</td>
</tr>
<tr>
<td>FREE ALL</td>
<td>Until all locks are freed</td>
</tr>
<tr>
<td>N/A</td>
<td>Not applicable to NOTIFY SUSPEND</td>
</tr>
</tbody>
</table>

**STATE**

The state or mode of the lock applied to the resource. Valid values are shown in Table 34-2 on page 34-5.

**ORIG.RSN**

The original reason for the suspension. The task remains suspended until all suspension causes are cleared. Valid values are shown in Table 34-5.

Table 34-5. Reason for Suspension - IRLM SUSPEND

<table>
<thead>
<tr>
<th>Reason</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>INTER SYSTEM</td>
<td>Intersystem communication required to resolve the lock request</td>
</tr>
<tr>
<td>IQ</td>
<td>Queued IRLM request</td>
</tr>
<tr>
<td>LOCAL CONTENTION</td>
<td>Local resource contention</td>
</tr>
<tr>
<td>LATCH CONT GENERIC</td>
<td>Generic IRLM latch contention</td>
</tr>
<tr>
<td>LATCH CONT MAIN</td>
<td>Main IRLM latch contention</td>
</tr>
</tbody>
</table>
Table 34-5. Reason for Suspension - IRLM SUSPEND (continued)

<table>
<thead>
<tr>
<th>Reason</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LATCH CONT NOTIFY</td>
<td>IRLM notify latch contention</td>
</tr>
<tr>
<td>LATCH CONT RESOURCE</td>
<td>IRLM resource latch contention</td>
</tr>
<tr>
<td>LATCH CONT WORKUNIT</td>
<td>IRLM work unit latch contention</td>
</tr>
<tr>
<td>LS</td>
<td>Local storage cannot be exceeded in cross-memory mode</td>
</tr>
<tr>
<td>NOTIFY MSG SENT</td>
<td>Intersystem message sending</td>
</tr>
<tr>
<td>RETAINED LOCK</td>
<td>Contention with a retained lock</td>
</tr>
</tbody>
</table>

PARENT
The parent token for explicit hierarchical locking.

HASH
The lock hash value.
The following fields are printed if both of the following conditions are satisfied:
• The OMEGAMON XE for DB2 PE subsystem is a member of a data sharing group.
• It is an IRLM suspension.

aaaaaaaaaaaaaaaaaaaaaaa
Stands for the lock attributes, which can be one or more of the following:
• MODIFY or NMODIFY
• GLOBAL or LOCAL
• P-LOCK or L-LOCK

XES PROP
An indicator whether or not IRLM propagated the request to XES. Possible values are Y(es) or N(o).

XES FORC
An indicator whether or not the lock was requested to be forced to XES. Possible values are Y(es) or N(o).

XES ASYN
An indicator whether or not IRLM sent the request asynchronously to XES. Possible values are Y(es) or N(o).
This field is only printed if XES PROP=Y.
Lock Suspension Events - Lock, Unlock, Change, and Notify Resume

The format of the data for these events depends on whether these events occurred in a data sharing or non-data-sharing environment.

“Format of data specific to Lock, Unlock, Change, and Notify Resume (Data Sharing)” shows the format in a data sharing environment, “Format of data specific to Lock, Unlock, Change, and Notify Resume (Non-Data Sharing)” in a non-data-sharing environment.

Format of data specific to Lock, Unlock, Change, and Notify Resume (Data Sharing)

This is the sample format for data specific to Lock, Unlock, Change, and Notify Resume in a data sharing environment.

```
09:05:44.38289840 CHANGE N/P N/P
09:05:44.38199696 RESUME DURATION =COMMIT LATCH CONTENTION=N
STATE =X IRLM QUEUED REQ =N
RESUME RSN=NORMAL GLOBAL CONT. =Y*
XES PROP =Y NOTIFY MSG SENT =N
XES FORC =N bbbbbbbbbbbbbbb
XES ASYN =Y RETAINED LOCK =N
Grm
PARENT =X‘7F5E64E0’
HASH =X‘00113406’
```

Format of data specific to Lock, Unlock, Change, and Notify Resume (Non-Data Sharing)

This is the sample format for data specific to Lock, Unlock, Change, and Notify Resume in a non-data-sharing environment.

```
SUSP.TIME =ss.nnnnnn LOCAL CONTENTION=Y
RESUME RSN=xxxxxxxx LATCH CONTENTION=Y*
```

Description of individual fields

The individual fields have the following meaning:

**SUSP.TIME**
The duration of the suspension.

**DURATION**
The length of time the lock is held. For a list of possible values, refer to Table 34-4 on page 34-19.

**STATE**
The state or mode of the lock applied to the resource. For a list of possible values, refer to Table 34-2 on page 34-5.

**RESUME RSN**
The reason for resumption. Valid values are shown in Table 34-6.

```
<table>
<thead>
<tr>
<th>Reason</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NORMAL</td>
<td>The suspended task resumed normally when the resource became available.</td>
</tr>
<tr>
<td>DEADLOCK</td>
<td>The suspended task resumed after a deadlock.</td>
</tr>
<tr>
<td>TIMEOUT</td>
<td>The suspended task resumed when a preset time interval expired.</td>
</tr>
<tr>
<td>IDENTIFY</td>
<td>The suspended task is resumed after an identify call to IRLM.</td>
</tr>
</tbody>
</table>
```

Table 34-6. Reason for Resume - IRLM Requests
**Locking Activity - Trace**

**XES PROP**
An indicator whether or not IRLM propagated the request to XES. Possible values are Y(es) or N(o).

**XES FORC**
An indicator whether or not the lock was requested to be forced to XES. Possible values are Y(es) or N(o).

**XES ASYN**
An indicator whether or not IRLM sent the request asynchronously to XES. Possible values are Y(es) or N(o).

This field is only printed if XES PROP=Y.

**aaaaaaaaaaaaaaaaaaaaa**
Stands for the lock attributes. It can be one or more of the following:
- MODIFY or NMODIFY
- GLOBAL or LOCAL
- P-LOCK or L-LOCK

For example:

**PARENT**
The parent token for explicit hierarchical locking.

**HASH**
The lock hash value.

**Description of suspension fields**

A request can be suspended for several reasons. For example, the original reason may have been an IRLM latch contention, then the request may first have hit local contention and, after it was resolved, global level contention. The fields in the right block show whether or not a particular reason for suspension was encountered, which is indicated by Y(es) or N(o). The original reason is marked with an asterisk (*).

**LOCAL CONTENTION**
The local resource contention.

**LATCH CONTENTION**
The IRLM latch contention.

**IRLM QUEUED REQ**
The IRLM queued request. This request is only valid for IRLM suspensions.

**GLOBAL CONT.**
The global contention. Intersystem communication is required to resolve the lock request. This reason applies to data sharing environments only.

**NOTIFY MSG SENT**
Intersystem message sending. This reason only applies to data sharing environments and IFCID 44 suspensions.

**bbbbbbbbbbbbbbbbbb**
Only applies if it is an IRLM suspension and the global contention is hit (GLOBAL CONT=Y). If these conditions are satisfied, it can be one of the following values:
Locking Activity - Trace

IRLM GLOBAL CONT
The request hit IRLM global resource contention.

XES GLOBAL CONT
The request hit XES global resource contention.

FALSE/SYNC-ASYNC
This can be one of the following:

FALSE CONT
The request is a false contention (shown if QW0045W8 is ON)

SYNC-TO-ASYNC CONV
The request is a sync-to-async conversion (shown if QW0045W8 is OFF)

RETAINED LOCK
Indicates whether there was contention with a retained lock.
Lock Suspension Events - Query Suspend

This topic shows the format of data specific to Query Suspend for Lock Suspension events. It also describes the fields provided for this event.

Format of data specific to Query Suspend

This is an example of the format of the data for this event.

ORIG.RSN=xxxxx xxxxxxxxxx

Field description

This field shows the original reason for suspension. For a list of possible values, refer to Table 34-5 on page 34-19.
Lock Suspension Events - Query Resume

This topic shows the format of data specific to Query Resume for Lock Suspension events. It also describes the fields provided for this event.

Format of data specific to Query Resume

The data specific to this event is derived from the IFCIDs 44/45 and 213/214. This is the format of the data for this event:

```
SUSP.TIME = t. nnnnn  LOCAL CONTENTION = Y
RESUME RSN = xxxxxxxx  LATCH CONTENTION = Y
IRLM QUEUED REQ = N
```

Field description

This field shows the original reason for suspension. For a list of possible values, refer to “Format of data specific to Lock, Unlock, Change, and Notify Resume (Data Sharing)” on page 34-21.
Locking Activity - Trace

Lock Suspension Events - Drain Suspend

This topic shows the format of data specific to Drain Suspend for Lock Suspension events. It also describes the fields provided for this event.

Format of data specific to Drain Suspend

This is the format of the data for this event:

```
CLAIM NO=nnnnn CLASS=xxxxx
```

Field description

The individual fields have the following meaning:

CLAIM NO

The number of claims held on this resource.

CLASS

The claim class. Valid values are shown in Table 34-7.

Table 34-7. Claim Classes - DRAIN SUSPEND

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS</td>
<td>Cursor stability read</td>
</tr>
<tr>
<td>RR</td>
<td>Repeatable read</td>
</tr>
<tr>
<td>WRITE</td>
<td>Write</td>
</tr>
</tbody>
</table>
Lock Suspension Events - Drain Resume

This topic shows the format of data specific to Drain Resume for Lock Suspension events. It also describes the fields provided for this event.

The data specific to this event is derived from the IFCDs 215 and 216. The format of the data for this event is shown in "Format of data specific to Drain Resume".

Format of data specific to Drain Resume

This is the format of the data for this event:

SUS.TIME=s.nnnnn CLASS =xxxxx
RESM.RSN=xxxxxxxx CLAIM NO=nnnnn

Field description

The individual fields have the following meaning:

**SUS.TIME**

The duration of the suspension.

**CLASS**

The claim class. Valid values are shown in [Table 34-7 on page 34-26](#).

**RESM.RSN**

The reason for resumption. Valid values are shown in the following table.

*Table 34-8. Reason for Resume - DRAIN RESUME*

<table>
<thead>
<tr>
<th>Reason</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NORMAL</td>
<td>The suspended task resumed normally when the resource became available.</td>
</tr>
<tr>
<td>TIMEOUT</td>
<td>The suspended task resumed when a preset time interval expired.</td>
</tr>
</tbody>
</table>

**CLAIM NO**

The number of claims held on this resource.
Lock Suspensions Events - Latch Suspend

This topic shows the format of data specific to Latch Suspend for Lock Suspension events. It also describes the fields provided for this event.

Format of data specific to Latch Suspend

This is the format of the data for this event:

```
TYPE=xxxxxxxxx
```

Field description

The field shown represents the type of the latch. It can have one of the following values:

Table 34-9. Latch Types - LATCH SUSPEND

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SHARED</td>
<td>S latch</td>
</tr>
<tr>
<td>EXCLUSIVE</td>
<td>X latch</td>
</tr>
</tbody>
</table>
Lock Suspension Events - Latch Resume

This topic shows the format of data specific to Latch Resume for Lock Suspension events. It also describes the fields provided for this event.

Format of data specific to Latch Resume

This is the format of the data for this event:

\[ \text{SUS.TIME} = \text{.nnnnn} \quad \text{TYPE} = \text{xxxxxxxxx} \]
\[ \text{STATUS} = \text{xxxxxxxxx} \]

Field description

The individual fields have the following meaning:

**SUS.TIME**

The duration of the suspension.

**TYPE**

The type of latch. Valid values are shown in Table 34-9 on page 34-28.

**STATUS**

The latch status. It can have one of the following values:

<table>
<thead>
<tr>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NORMAL</td>
<td>Normal completion of a page latch wait.</td>
</tr>
<tr>
<td>CANCELLED</td>
<td>The page latch wait was canceled before the latch was obtained. For example,</td>
</tr>
<tr>
<td></td>
<td>the agent representing the latch was abnormally terminated during a page</td>
</tr>
<tr>
<td></td>
<td>latch wait.</td>
</tr>
</tbody>
</table>

Table 34-10. Latch Status - LATCH RESUME
Lock Detail Trace Data

The lock detail trace describes all locking events in a DB2 system. It includes those that can be viewed in suspension, timeout, or deadlock traces. This trace gives you a global view of the entire locking activity in the system.

You determine which locking events you want to see in a lock detail trace. You do this in the TRACE command by specifying the TYPE identifier in the INCLUDE and EXCLUDE options.

The lock detail trace is produced if level DETAIL is specified on the TRACE command and if there is at least one IFCID in the input data set that satisfies the FROM and TO, and INCLUDE or EXCLUDE criteria.

The following sections show the sample layout of a Lock Detail trace and describe the various fields of the trace.

"Example of a Lock Detail Trace" on page 34-31
This topic shows an example of a Lock Detail trace.

"Lock Detail Trace - Lock Summary" on page 34-32
This section shows the event specific data for Lock Summary.

"Lock Detail Trace - Lock, Unlock, and Change Requests" on page 34-34
This section shows the event specific data for Lock, Unlock, and Change Requests.

"Lock Detail - Query Requests" on page 34-37
This section shows the event specific data for Query Requests.

"Lock Detail - Claim Acquire, Change, and Release" on page 34-38
This section shows the event specific data for Claim Acquire, Change, and Release.

"Lock Detail - Drain Request, Pseudo, and Release" on page 34-40
This section shows the event specific data for Drain Request, Pseudo, and Release.

"Lock Detail - Lock Avoidance" on page 34-42
This section shows the event specific data for Lock Avoidance.

"Lock Detail - P-Lock Requests" on page 34-43
This section shows the event specific data for P-Lock requests.

"Lock Detail - Notify Request" on page 34-46
This section shows the event specific data for Notify request.

"Lock Detail - Lock Escalatn" on page 34-47
This section shows the event specific data for Lock Escalatn.
Example of a Lock Detail Trace

This topic shows an example of a Lock Detail trace.

"Lock Detail trace example" shows a sample Lock Detail trace, produced by the following command:

```
...
LOCKING
   TRACE
       LEVEL (DETAIL)
...
```

Lock Detail trace example

This is the sample layout of a Lock Detail trace.

```
LOCATION: OMPDA21 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-1
GROUP: N/P LOCKING TRACE - DETAIL REQUESTED FROM: NOT SPECIFIED
MEMBER: N/P TO: NOT SPECIFIED
SUBSYSTEM: DA21 ACTUAL FROM: 02/14/11 19:45:51.41
DB2 VERSION: V10 SCOPE: MEMBER PAGE DATE: 02/14/11

PRIMAUTH CORNAME CONNTYPE ORIGAUTH CORNNAME INSTANCE PLANNAME CONNECT EVENT TIMESAMP --- LOCK RESOURCE ---
--- -------- --------- ----------------------- ----------------------------------------
19:45:53.99629213 LOCK SKPT DURATION=COMMIT STATE=S REQUEST RSN CODE= 0 RTNCD= 0
CTKN=18861A9002FAE19 HASH =X'00003280'
COLL(NEX)= X'122344556677889900112234455667788'
PKID(NEX)= X'12234455667788'

LOCKING TRACE COMPLETE
```
Lock Detail Trace - Lock Summary

This section shows the event specific data for Lock Summary.

The format of Lock Summary data

The following sample shows the format of the data for this event.

```
MAX PAGE & ROW LOCKS= 11  LOCKAV=YES
SHARED ESCAL= 0  EXCLUS.ESCAL= 0

MAX PAGE & ROW LOCKS= 1  LOCKAV=NO
TABLESPACE TYPE=UNSEGMENTED  SIZE=PAGE
TABLES WITH ESCALATIONS=XXXXXX
MAX STATE=XXXXX  PRE-ESCAL.STATE=XXXXX
```

Field description

The individual fields have the following meaning:

**MAX PAGE & ROW LOCKS**

The maximum number of page or row locks across all table spaces held concurrently for the thread.

If IFCID 020 is not present, N/A is printed in this field.

**LOCKAV**

Indicates if lock avoidance techniques are used within this unit of work across all table spaces. Possible values are Y(es) or N(o).

If IFCID 218 is not present, N/A is printed in this field.

**SHARED ESCAL**

The number of escalations to shared mode for the thread:

- For segmented table spaces, the number of tables that have escalated
- For partitioned table spaces using selective partition locking (SPL), the number of partitions that have escalated
- For simple and partitioned table spaces, the number of table spaces that have escalated

If IFCID 020 is not present, N/A is printed in this field.

**EXCLUS.ESCAL**

The number of escalations to exclusive mode for the thread:

- For segmented table spaces, the number of tables that have escalated
- For partitioned table spaces using selective partition locking (SPL), the number of partitions that have escalated
- For simple and partitioned table spaces, the number of table spaces that have escalated

If IFCID 020 is not present, N/A is printed in this field.

Fields that are printed once for each table space

The following fields are printed once for each table space:

**MAX PAGE & ROW LOCKS**

The maximum number of page or row locks per table space held concurrently by the thread.

If IFCID 020 is not present, N/A is printed in this field.
LOCKAV
Indicates if lock avoidance techniques are used for this table space.
Possible values are Y(es) or N(o).
If IFCID 218 is not present, N/A is printed in this field.

TABLE SPACE TYPE
The table space type:

SIMPLE
Simple table spaces

SEGMENTED
Segmented table spaces

PARTITIONED
Partitioned table spaces

PARTIT.-SPL
Partitioned table spaces using selective partition locking (SPL)
If IFCID 020 is not present, the table space type is not printed.

SIZE
The lock size used, which can be one of the following:
• PAGE
• ROW
• TABLE

TABLES WITH ESCALATIONS
The number of tables within the table space for which escalations occurred.
This field is only printed for segmented table spaces or partitioned table spaces using SPL.
If IFCID 020 is not present, N/A is printed in this field.

MAX STATE
The highest lock state for the table space. This field is only printed for simple table spaces or partitioned table spaces not using SPL.
If IFCID 020 is not present, N/A is printed in this field.
Possible values are shown in Table 34-11.

<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IS</td>
<td>Intent share</td>
</tr>
<tr>
<td>IX</td>
<td>Intent exclusive</td>
</tr>
<tr>
<td>S</td>
<td>Share</td>
</tr>
<tr>
<td>U</td>
<td>Update</td>
</tr>
<tr>
<td>SIX</td>
<td>Share with intent exclusive</td>
</tr>
</tbody>
</table>

Table 34-11. Lock State

PRE-ESCAL. STATE
The lock state before escalations. A list of values is shown in Table 34-11. If no escalation occurred, NO ESCALATIONS is printed.
This field is only printed for simple table spaces or partitioned table spaces not using SPL.
If IFCID 020 is not present, N/P is printed.
Lock Detail Trace - Lock, Unlock, and Change Requests

This section shows the event specific data for Lock, Unlock, and Change Requests.

Format of Lock, Unlock, and Change Requests

The following sample shows the format of the data for these events.

\[
\begin{align*}
\text{DURATION} & = xxxxxxxx \\
\text{STATE} & = xxxx \\
\text{PROP} & = x \\
\text{RSN CODE} & = xxxxxxxx \\
\text{RTNCD} & = xx \\
\text{XES FORC} & = x \\
\text{XES ASYN} & = x \\
\text{PARENT} & = xxxxxxxx \\
\text{CACHE} & = xxx \\
\text{OWNER} & = xxxxxxxx \\
\text{HASH} & = x'hhhhhhhh' \\
\end{align*}
\]

Field description

The individual fields have the following meaning:

DURATION
The length of time the lock is held. Valid values are shown in Table 34-12.

<table>
<thead>
<tr>
<th>Duration</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>INTEREST</td>
<td>Duration used for P-Locks</td>
</tr>
<tr>
<td>MANUAL</td>
<td>Varies depending on the ISOLATION parameter</td>
</tr>
<tr>
<td>MANUAL+1</td>
<td>Temporary change of consistency level from CS to RR during bind and DDL</td>
</tr>
<tr>
<td>COMMIT</td>
<td>Until commit</td>
</tr>
<tr>
<td>COMMIT+1</td>
<td>Past commit; applies to locks needed to maintain the position for a cursor opened WITH HOLD</td>
</tr>
<tr>
<td>ALLOCATN</td>
<td>Until deallocation</td>
</tr>
<tr>
<td>PLAN</td>
<td>For the duration of the plan</td>
</tr>
<tr>
<td>UTILITY</td>
<td>For the duration of the utility execution</td>
</tr>
<tr>
<td>FREE ALL</td>
<td>Until all locks are freed</td>
</tr>
<tr>
<td>N/A</td>
<td>Not applicable for NOTIFY SUSPEND</td>
</tr>
</tbody>
</table>

The duration controls when locks are released. A lock is usually only released when an agent makes an unlock request with a duration longer, or equal to, the longest lock duration the agent specified for the resource.

You can increase lock durations using either a lock request or a change request.

STATE
The state or mode of the lock applied to the resource. Valid values are shown in Table 34-13.

<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IS</td>
<td>Intent share</td>
</tr>
<tr>
<td>IX</td>
<td>Intent exclusive</td>
</tr>
<tr>
<td>S</td>
<td>Share</td>
</tr>
<tr>
<td>U</td>
<td>Update</td>
</tr>
<tr>
<td>SIX</td>
<td>Share with intent exclusive</td>
</tr>
<tr>
<td>NSU</td>
<td>Nonshared update</td>
</tr>
</tbody>
</table>
Table 34-13. Lock State-IRLM Requests (continued)

<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>Exclusive</td>
</tr>
<tr>
<td>N/A</td>
<td>Not applicable for NOTIFY SUSPEND</td>
</tr>
</tbody>
</table>

**RTNCD**

The return code issued in response to the request. The possible return codes are shown in Table 34-14.

Table 34-14. Return Codes-IRLM Requests

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Successful completion</td>
</tr>
<tr>
<td>4</td>
<td>Successful completion, lock state unchanged</td>
</tr>
<tr>
<td>8</td>
<td>Unsuccessful completion, system error</td>
</tr>
<tr>
<td>12</td>
<td>Unsuccessful completion, logic error in request</td>
</tr>
<tr>
<td>16</td>
<td>Unsuccessful completion, request specification not valid</td>
</tr>
</tbody>
</table>

**RSN CODE**

The reason code issued in response to the request. The reason code is not applicable for lock avoidance.

**Fields that are only printed if the DB2 subsystem is a member of a data sharing group**

The remaining fields are only printed if the DB2 subsystem is a member of a data sharing group.

aaaaaaaabbbbbbcdddeeeeee

Stands for the lock attributes, which can be:
- MODIFY or NMODIFY
- GLOBAL or LOCAL
- P-LOCK or L-LOCK

**PARENT**

The parent lock token if one was specified for explicit hierarchical locking.

The field is only printed for LOCK REQUESTs.

If this field is not 0, the request applies to a child of a parent that has already been locked.

**CACHE**

The cached state of a P-Lock. For the state values, refer to Table 34-13 on page 34-34.

This field is only applicable and printed for page set and partition P-Locks.

**OWNER**

The member name of the owner of a retained lock that caused this request to be denied and the owner of the lock that caused this request to time out. If neither of these conditions exist, this field is not printed.

**HASH**

The lock hash value.
Locking Activity - Trace

**XES PROP**
An indicator whether or not IRLM propagated the request to XES. Possible values are Y(es) or N(o).

**XES FORC**
An indicator whether or not the lock was requested to be forced to XES. Possible values are Y(es) or N(o).

**XES ASYN**
An indicator whether or not IRLM sent the request asynchronously to XES. Possible values are Y(es) or N(o).
This field is only printed if XES PROP=Y.
Lock Detail - Query Requests

This section shows the event specific data for Query Requests.

Format of Query Requests

This is an example of the format of the data for this event.

RSN CODE=xxxxxxxxx RTNCD=xxx

For an explanation of the individual fields refer to Table 34-14 on page 34-35.
**Locking Activity - Trace**

**Lock Detail - Claim Acquire, Change, and Release**

This section shows the event specific data for Claim Acquire, Change, and Release.

**Format of Claim Acquire, Change, and Release**

This is an example of the format of the data for this event.

DURATION=xxxxxxxxx
CLASS=xxxxx
RSN CODE=xxxxxxxxx
RTNCD=x

**Field description**

The individual fields have the following meaning:

**DURATION**

The duration of the claim. The values for this field are shown in Table 34-15.

<table>
<thead>
<tr>
<th>Duration</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMIT</td>
<td>Until commit</td>
</tr>
<tr>
<td>COMMIT+1</td>
<td>Past commit; applies to locks needed to maintain the position for a cursor opened WITH HOLD</td>
</tr>
<tr>
<td>ALLOCATN</td>
<td>Until deallocation</td>
</tr>
</tbody>
</table>

**CLASS**

The claim class of the claim request. The values for this field are shown in Table 34-16.

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS</td>
<td>Cursor stability read</td>
</tr>
<tr>
<td>RR</td>
<td>Repeatable read</td>
</tr>
<tr>
<td>WRITE</td>
<td>Write</td>
</tr>
</tbody>
</table>

**RSN CODE**

The reason code issued in response to the request. The values for this field are shown in Table 34-17.

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Successful claim</td>
</tr>
<tr>
<td>00C90080</td>
<td>Unsuccessful claim, resource is started</td>
</tr>
<tr>
<td>00C90081</td>
<td>Unsuccessful claim, resource is stopped</td>
</tr>
<tr>
<td>00C90082</td>
<td>Unsuccessful claim, resource is used by a utility</td>
</tr>
<tr>
<td>00C90083</td>
<td>Unsuccessful claim, resource is used by a utility that allows R/O access only</td>
</tr>
<tr>
<td>00C90086</td>
<td>Unsuccessful claim, resource is started for utility-only access</td>
</tr>
<tr>
<td>00C90088</td>
<td>Unsuccessful claim, deadlock</td>
</tr>
<tr>
<td>00C9008E</td>
<td>Unsuccessful claim, timeout on drain lock</td>
</tr>
<tr>
<td>00C90092</td>
<td>Unsuccessful claim, IRLM out-of-storage condition</td>
</tr>
</tbody>
</table>
Locking Activity - Trace

Table 34-17. Reason Codes-Claim Requests (continued)

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00C90093</td>
<td>Unsuccessful claim, IRLM error</td>
</tr>
<tr>
<td>00C90097</td>
<td>Unsuccessful claim, resource has an image copy pending</td>
</tr>
<tr>
<td>00C900A0</td>
<td>Unsuccessful claim, resource has recovery pending</td>
</tr>
<tr>
<td>00C900A3</td>
<td>Unsuccessful claim, resource has a check pending</td>
</tr>
</tbody>
</table>

RTNCD

The return code issued in response to the request. The values for this field are shown in Table 34-18.

Table 34-18. Return Codes-Claim Requests

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Successful completion</td>
</tr>
<tr>
<td>4</td>
<td>Logical claim needed</td>
</tr>
<tr>
<td>8</td>
<td>Unsuccessful completion</td>
</tr>
</tbody>
</table>
Lock Detail - Drain Request, Pseudo, and Release

This section shows the event specific data for Drain Request, Pseudo, and Release.

Format of Drain Request, Pseudo, and Release

This is an example of the format of the data for this event.

STATE =xxxxx CLASS=xxxxx
RSN CODE=xxxxxxxx RTNCD=x

Field description

The individual fields have the following meaning:

STATE
The lock state. It is only applicable to DRAIN REQUEST. Possible values are shown in Table 34-19.

<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IX</td>
<td>Intent exclusive</td>
</tr>
<tr>
<td>X</td>
<td>Exclusive</td>
</tr>
</tbody>
</table>

CLASS
The claim class of the drain request. The values for this field are shown in Table 34-20.

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS</td>
<td>Cursor stability read</td>
</tr>
<tr>
<td>RR</td>
<td>Repeatable read</td>
</tr>
<tr>
<td>WRITE</td>
<td>Write</td>
</tr>
</tbody>
</table>

RSN CODE
The reason code issued in response to the request. The values for this field are shown in Table 34-21.

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Successful claim</td>
</tr>
<tr>
<td>00C90088</td>
<td>Unsuccessful claim, deadlock</td>
</tr>
<tr>
<td>00C9008E</td>
<td>Unsuccessful claim, timeout</td>
</tr>
<tr>
<td>00C90092</td>
<td>Unsuccessful claim, IRLM out-of-storage condition</td>
</tr>
<tr>
<td>00C90093</td>
<td>Unsuccessful claim, IRLM error</td>
</tr>
</tbody>
</table>

RTNCD
The return code issued in response to the request. The values for this field are shown in Table 34-22.

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Successful completion</td>
</tr>
</tbody>
</table>
Locking Activity - Trace

Table 34-22. Return Codes-Drain Requests (continued)

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>Unsuccessful completion</td>
</tr>
</tbody>
</table>
Locking Activity - Trace

Lock Detail - Lock Avoidance
This section shows the event specific data for Lock Avoidance.

This event does not have event-specific data.
Lock Detail - P-Lock Requests

This section shows the event specific data for P-Lock requests.

P-Lock requests include the following events:

- **Page set or partition P-Lock request or page set or partition P-Lock negotiation request**
  
  These P-Locks track inter-DB2 interest on a linear page set (table space or index) or a partition of a partitioned page set. The cached state of the page set or partition P-Lock tells DB2 which data sharing protocols must be used to maintain inter-DB2 buffer coherency for the page set or partition. For example, a cached state of IS tells DB2 that whenever a page belonging to that page set or partition is read into the buffer pool, the page must be registered to the coupling facility for cross-invalidation purposes. If the cached state were SIX, the coupling facility page registration would not be necessary. Normally the P-Lock is held by DB2 in the cached state. The P-Lock state determines whether or not the page set or partition is GBP-dependent:
  
  - If the page set or partition P-Lock is held in S or X, then the page set or partition is not GBP-dependent.
  - Otherwise, the page set or partition is GBP-dependent.

- **Page P-Lock request or page P-Lock negotiation request**

  These P-Locks preserve the inter-DB2 cached page (buffer) coherency when subpage concurrency protocols are used and the page set or partition is actively R/W shared between two or more DB2 systems. The most common cases of subpage concurrency are row-level locking and type-1 index minipages.

**Note**: Page P-Locking can add a significant overhead to data sharing if inter-DB2 workloads are not properly balanced. Class 21 is added to monitor these events without having to use the costly Class 7. However, page P-Lock events are recorded in Class 7 as well. Therefore, if Class 7 and 21 are both active, two records are reported for the same event.

The format of the data for these events is shown in ["Format of Page Set or Partition P-Lock Requests"] and ["Format of Page P-Lock Requests.

Format of Page Set or Partition P-Lock Requests

The following example shows the format of the data for this event.

```
REQUEST=xxxxxx OBJECT=xxxxxxxxxxxx
MEMBER =xxxxxxxx REQUESTED STATE =xxxxx
OLD STATE=xxxxx OLD CACHED STATE=xxxxx
NEW STATE=xxxxx NEW CACHED STATE=xxxxx
```

Format of Page P-Lock Requests

The following example shows the format of the data for this event.

```
REQUEST=xxxxxx OBJECT=xxxxxxxxxxxx
MEMBER =xxxxxxxx REQUESTED STATE =xxxxx
OLD STATE=xxxxx NEW STATE=xxxxx
```

Field description

Here is a description of the field labels shown in the previous examples:
REQUEST
The IRLM request type, which can be one of the following:
- LOCK
- UNLOCK
- CHANGE
- EXIT

OBJECT
The DB2 object type, which can be one of the following:
- TABLESPACE
- INDEXSPACE
- DATA PAGE
- HEADER PAGE
- INDEX PAGE
- SPACE MAP PAGE

MEMBER
A DB2 member name that depends on the request type:
- For exit requests, the name of the DB2 member in conflict with this member’s currently held P-Lock state.
- For lock, unlock, and change requests, for which P-Lock is rejected, the name of the DB2 member in conflict with this request.

REQUESTED STATE
The requested lock state. It only applies to lock, change, and exit requests.
For exit requests, this is the P-Lock state requested by the member causing the P-Lock exit of this member. If the request from the other member was not in conflict with the state of this member, this field shows NH.
The values for this field are shown in Table 34-23.

<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NH</td>
<td>Not held</td>
</tr>
<tr>
<td>IS</td>
<td>This DB2 has R/O interest on the page set or partition and one or more other DB2s in the group have R/W interest</td>
</tr>
<tr>
<td>IX</td>
<td>This DB2 has R/W interest on the page set or partition, one or more other DB2s in the group have R/O interest, and one or more can also have R/O interest</td>
</tr>
<tr>
<td>S</td>
<td>This DB2 has R/O interest on the page set or partition and no other DB2 in the group has R/W interest but one or more can have R/O interest</td>
</tr>
<tr>
<td>SIX</td>
<td>This DB2 has R/W interest on the page set or partition and no other DB2 in the group has R/W interest but one or more can have R/O interest</td>
</tr>
<tr>
<td>NSU</td>
<td>Nonshared update</td>
</tr>
<tr>
<td>X</td>
<td>This DB2 has R/W interest on the page set or partition and no other DB2 in the group has any interest</td>
</tr>
<tr>
<td>RD</td>
<td>Request denied</td>
</tr>
</tbody>
</table>

OLD STATE
The previously held P-Lock state.
The values for this field are shown in Table 34-23.
NEW STATE
The newly held P-Lock state.
The values for this field are shown in Table 34-23 on page 34-44.

OLD CACHED STATE
The previous P-Lock cached state.
The values for this field are shown in Table 34-23 on page 34-44.

NEW CACHED STATE
The new P-Lock cached state.
The values for this field are shown in Table 34-23 on page 34-44.

Stands for the P-Lock attributes, which can be one or more of the following:
- UNCONDITIONAL or CONDITIONAL.
- RESTART or NONRESTART. Such a request instructs IRLM to convert a retained lock held by the DB2 system into an active lock. If the requested lock is not retained, IRLM grants the request as normal.
- MODIFY or NONMODIFY.
Lock Detail - Notify Request

This section shows the event specific data for Notify request.

In some cases, DB2 data sharing uses the IRLM notify request to maintain non-buffer pool cache coherency between DB2 systems in the group. Examples of a notify request usage are DBD coherency and High Used RBA (HURBA) for a data set.

Format of Data Specific to NOTIFY REQUEST

This is an example of the format of the data for this event.

\[
\text{TYPE} = \text{x.x.x.x.x.x} \quad \text{STATE} = \text{x.x.x.x} \\
\text{WAIT} = \text{xxx} \quad \text{HOLDERS} = \text{xxxxx}
\]

Field description

Here is a description of the field labels shown in the previous example:

**TYPE**  The type of notify operation. Possible values are SEND or RECEIVE.

**STATE**  The lock state. For a list of possible values, refer to Table 34-13 on page 34-34.

If this field contains one of the listed values, only those lock holders owning the lock in the specified state are notified.

*N/A* in this field means that the notify message is sent to all DB2 systems holder of the lock, regardless of the state they hold it in.

This field is not applicable or printed if **TYPE**=RECEIVE.

**WAIT**  Indicates if the request is synchronous, that is, suspended until all responses are received, in which case **WAIT**=YES is printed, or asynchronous, that is, **WAIT**=NO.

**HOLDERS**  The number of holders notified.

This field is not applicable or printed if **TYPE**=RECEIVE.
Lock Detail - Lock Escalation

This section shows the event specific data for Lock Escalation.

This trace shows details of a lock escalation. It is shown when data from IFCID 337 is present in the input data.

Format of the Lock Escalation

This sample shows the format of the data specific to Lock Escalation.

```
STATE = IS
NUMLOCKS = 815
STMTNO = 4711
STMTINFO = XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
STMTID = X'XXXXXXXXXXXXXXXXXXXXX'
COLLID = COLLECTIONXXXXXXXXX1XXXXXXXXX2XXXXXXXXX3
XXXXXXXXX4XXXXXXXXX5XXXXXXXXX6
XXXXXXXXX7XXXXXXXXX8XXXXXXXXX9
XXXXXXXXX0XXXXXXXXX1XXXXXXXXX2
XXXXXXXXX3
XXXXXXXXX4

PACKAGE = PACKAGEXX1XXXXXXXXX2XXXXXXXXX3
XXXXXXXXX4XXXXXXXXX5XXXXXXXXX6
XXXXXXXXX7XXXXXXXXX8XXXXXXXXX9
XXXXXXXXX0XXXXXXXXX1XXXXXXXXX2
XXXXXXXXX3
```

Field description

**STATE**

The state or mode of the lock applied to the resource. Valid values are shown in Table 34-24.

<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IS</td>
<td>Intent share</td>
</tr>
<tr>
<td>IX</td>
<td>Intent exclusive</td>
</tr>
<tr>
<td>S</td>
<td>Share</td>
</tr>
<tr>
<td>U</td>
<td>Update</td>
</tr>
<tr>
<td>SIX</td>
<td>Shared intent exclusive</td>
</tr>
<tr>
<td>X</td>
<td>Exclusive</td>
</tr>
</tbody>
</table>

**NUMLOCKS**

Number of held lower level locks that were released by escalation.

**STMTNO**

Statement number.

**STMTINFO**

The waiter statement information. Possible values are:

- **STATIC**
  The statement is of type static

- **DYNAMIC**
  The statement is of type dynamic.

- **NONE**
  No statement ID, no type.

**STMTID**

The waiter statement ID.
Locking Activity - Trace

COLLID
Collection ID.

PACKAGE
Package name.
Chapter 35. The Locking File Data Set

The locking file data set creates a sequential data set of formatted DB2 locking detail records that can be loaded into the OMEGAMON XE for DB2 PE performance database using the DB2 load utility.

Use the performance database to produce tailored reports using a reporting facility such as Query Management Facility (QMF).

The locking file data set contains a record for each occurrence of the following events:

- A LOCK, UNLOCK, CHANGE, or QUERY request processed by DB2
- A request to acquire a claim, change a claim duration, or release a claim
- A request to release a drain on a claim class
- Whenever lock avoidance is successful

The output of the FILE command is a sequential variable blocked data set.

The content of the output data set is determined by the FILE command options you specify, and by the input SMF/GTF records processed.

Descriptions of the layouts of these records can be found in the RKO2SAMP library. The member name is DGOLDFIL.

**Note:** For an introduction to the Locking report set and general locking information refer to the *Reporting User’s Guide*.
Accounting Report
Part 7. The Record Trace Report Set

These topics provide information about the record trace reports.

**Note:** For an introduction to the Record Trace report set and general Record Trace information refer to the Reporting User’s Guide. It also provides information on input to Record Trace reports.

- **Chapter 36, “Record Headers,” on page 36-1**
  Records written in a record trace report are prefixed by a header. The header is rewritten if any of the header information changes.

- **Chapter 37, “The Summary Record Trace,” on page 37-1**
  The summary record trace lists all records in the same sequence as an input data set.

- **Chapter 38, “The Short and Long Record Traces,” on page 38-1**
  The short and long record traces are similar. The short record trace reports non-serviceability data from records which are used by other subcomponents of the batch component. Serviceability records and fields are not printed on the short record trace. Only the occurrence of large records such as statistics, accounting, and system parameters is shown.

- **Chapter 39, “Dump Record Trace,” on page 39-1**
  The dump record trace lists all data from selected records of an input data set.

- **Chapter 40, “IFCID Record Blocks,” on page 40-1**
  This topic describes the Instrumentation Facility Component Identifier (IFCID) record trace blocks. The description within each block is presented in alphabetical order.

- **Chapter 41, “The Record Trace File Data Set and Output Records,” on page 41-1**
  The record trace file data set is a sequential data set of formatted records suitable for loading into the performance database using the DB2 load utility and from which reports can be produced using a reporting facility such as Query Management Facility (QMF).
Chapter 36. Record Headers

Records written in a record trace report are prefixed by a header. The header is rewritten if any of the header information changes.

<table>
<thead>
<tr>
<th>PRIMAUTH</th>
<th>ORIGAUTH</th>
<th>PLANNAME</th>
<th>CONNECT</th>
<th>INSTANCE</th>
<th>END_USER</th>
<th>WS_NAME</th>
<th>RECORD TIME</th>
<th>DESTNO</th>
<th>IFC</th>
<th>DESCRIPTION</th>
<th>TRANSACT</th>
<th>DATA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>SOF</td>
<td>BATCH</td>
<td>B1BAA2A382C5</td>
<td>'ANDREW'</td>
<td>'ANDREWS_PC'</td>
<td>'BLANK'</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SOF</td>
<td>TSO</td>
<td>23:48:01.86220375</td>
<td>41</td>
<td>1</td>
<td>3 ACCOUNTING</td>
<td>NETWORKID: Y61Y</td>
<td>LUNAME: STM4Y61Y</td>
<td>LUWSEQ: 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SOF</td>
<td>TSO</td>
<td>23:48:01.86220375</td>
<td>41</td>
<td>1</td>
<td>3 ACCOUNTING</td>
<td>NETWORKID: Y61Y</td>
<td>LUNAME: STM4Y61Y</td>
<td>LUWSEQ: 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SOF</td>
<td>TSO</td>
<td>23:48:01.86220375</td>
<td>41</td>
<td>1</td>
<td>3 ACCOUNTING</td>
<td>NETWORKID: Y61Y</td>
<td>LUNAME: STM4Y61Y</td>
<td>LUWSEQ: 1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Field description

The following information is reported in the trace header:

**PRIMAUTH**
The authorization ID under which the transaction is running. Derived from the DB2 field QWHCAID.

**ORIGAUTH**
The original authorization ID under which the transaction started. Derived from the DB2 field QWHCOPID.

**PLANNAME**
The DB2 plan name. Derived from the DB2 field QWHCPLAN.

**CONNECT**
The connection ID. Derived from the DB2 field QWHCCN.

**CORRNAME**
The correlation name. Derived from the DB2 field QWHCCV.

**CORRNMBR**
The correlation number. Derived from the DB2 field QWHCCV.

**INSTANCE**
The unique number assigned to a thread. Derived from the DB2 field QWHSLUUV.

**CONNTYPE**
The type of connection being used to interface with DB2. Derived from the DB2 field QWHATYP.

**END_USER**
User ID of the workstation end user. Derived from the DB2 field QWHCEUID.

**RECORD TIME**
The timestamp contained in the trace record. The format is hours, minutes, seconds, and hundred-millionths of a second. Derived from the DB2 field QWHSSTCK.

**TCB CPU TIME**
The CPU time stored in the trace record. The format is minutes, seconds, and hundred-millionths of a second. Derived from the DB2 field QWHUCPU.

**WS_NAME**
Name of the workstation. Derived from the DB2 field QWHCEUWN.
Record Trace - Headers

**DEST SEQ NO**
The destination sequence number. Derived from the DB2 field QWHSWSEQ.

**ACE**
The relative number of the agent control element address in the ACE cross-reference table. This table is printed at the end of each location for every trace specified. Derived from the DB2 field QWHSACE.

**IFCID**
The instrumentation facility component identification (DB2 trace record type). Derived from the DB2 field QWHSIID.

**DESCRIPTION**
A brief description of the IFCID record. The description indicates whether the record contains accounting, statistics, or performance data. For performance data, the description also indicates the event.

**TRANSACT**
Name of the workstation transaction. Derived from the DB2 field QWHCEUTX.

**DATA**
The data is printed in the standard hexadecimal dump format. The character format is on the right.
Chapter 37. The Summary Record Trace

The summary record trace lists all records in the same sequence as an input data set.

You can use this listing to check which records are in the DB2 instrumentation trace data. The short trace and long traces are normally too bulky for this purpose.

The summary record trace can be used with all the selection options such as INCLUDE and EXCLUDE.

The following command produces the summary record trace shown in "Example of the Summary Record Trace."

```plaintext
RECTRACE Trace
FROM (,17:38:00)
TO (,17:40:00)
LEVEL (SUMMARY)
```

Example of the Summary Record Trace

Here is an example of a Summary record trace.

```plaintext
LOCATION: STLEC1 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-1
GROUP: DSNCAT RECORD TRACE - SUMMARY REQUESTED FROM: ALL 17:38:00.00
MEMBER: V71A TO: DATES 17:40:00.00
SUBSYSTEM: V71A ACTUAL FROM: 01/30/10 17:38:00.83
DB2 VERSION: V10 PAGE DATE: 01/30/10

<table>
<thead>
<tr>
<th>ORIGAUTH</th>
<th>CORRNAME</th>
<th>CORRMNR</th>
<th>INSTANCE</th>
<th>CONTTYPE</th>
<th>END USER</th>
<th>RECORD TIME</th>
<th>TCB CPU TIME</th>
<th>DESTACE</th>
<th>IDC</th>
<th>FID</th>
<th>DESCRIPTION</th>
<th>TRANSACT DATA</th>
</tr>
</thead>
<tbody>
<tr>
<td>USRT014</td>
<td>B0A5B5E18F4B 'BLANK'</td>
<td>'BLANK'</td>
<td>USRT014</td>
<td>BATCH</td>
<td>17:38:00.00-0.839300</td>
<td>1090</td>
<td>1 62 DDL --&gt; 'BLANK'</td>
<td>NETWORKID: USIBMSY LUNAME: SYEC1DB2 LUWSEQ: 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>USRT014</td>
<td>T1240108 TSO 'BLANK'</td>
<td>'BLANK'</td>
<td>USRT014</td>
<td>T1240108</td>
<td>17:38:00.00-0.834593</td>
<td>1091</td>
<td>16 INSERT --&gt; 'BLANK'</td>
<td>NETWORKID: USIBMSY LUNAME: SYEC1DB2 LUWSEQ: 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DSNTEP3</td>
<td>'BLANK'</td>
<td>'BLANK'</td>
<td>DSNTEP3</td>
<td>'BLANK'</td>
<td>17:38:00.00-0.837502</td>
<td>1092</td>
<td>1 18 SCAN END &lt;-- 'BLANK'</td>
<td>NETWORKID: USIBMSY LUNAME: SYEC1DB2 LUWSEQ: 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>USRT013</td>
<td>B0A5B5F667C2 'BLANK'</td>
<td>'BLANK'</td>
<td>USRT013</td>
<td>BATCH</td>
<td>17:38:23.097392</td>
<td>1096</td>
<td>2 233 CALL STORED --&gt; 'BLANK'</td>
<td>NETWORKID: USIBMSY LUNAME: SYEC1DB2 LUWSEQ: 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>USRT013</td>
<td>T1240109 TSO 'BLANK'</td>
<td>'BLANK'</td>
<td>USRT013</td>
<td>T1240109</td>
<td>17:38:30.750538</td>
<td>1097</td>
<td>2 16 INSERT --&gt; 'BLANK'</td>
<td>NETWORKID: USIBMSY LUNAME: SYEC1DB2 LUWSEQ: 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLJBP147</td>
<td>'BLANK'</td>
<td>'BLANK'</td>
<td>PLJBP147</td>
<td>'BLANK'</td>
<td>17:38:31.251125</td>
<td>1098</td>
<td>2 233 CALL STORED --&gt; 'BLANK'</td>
<td>NETWORKID: USIBMSY LUNAME: SYEC1DB2 LUWSEQ: 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>USRT013</td>
<td>B0A5B5F667C2 'BLANK'</td>
<td>'BLANK'</td>
<td>USRT013</td>
<td>BATCH</td>
<td>17:38:52.0102268</td>
<td>1099</td>
<td>2 16 INSERT --&gt; 'BLANK'</td>
<td>NETWORKID: USIBMSY LUNAME: SYEC1DB2 LUWSEQ: 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>USRT013</td>
<td>B0A5B5F667C2 'BLANK'</td>
<td>'BLANK'</td>
<td>USRT013</td>
<td>BATCH</td>
<td>17:38:55.094477</td>
<td>1100</td>
<td>2 17 SEQ. SCAN --&gt; 'BLANK'</td>
<td>NETWORKID: USIBMSY LUNAME: SYEC1DB2 LUWSEQ: 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

© Copyright IBM Corp. 1985, 2013
Record Trace - Summary

0.1236778 17:38:56.72342318 0.17088218 0.13966363 0.07857034 0.07886559 0.09435964 0.06625079
1107 2 16 INSERT --> 'BLANK' SCAN BEGIN NETWORKID: USIBMSY LUNAME: SYEC1DB2 LUWSEQ: 1
1108 2 16 INSERT --> 'BLANK' SCAN BEGIN NETWORKID: USIBMSY LUNAME: SYEC1DB2 LUWSEQ: 1
1109 2 233 CALL STORED 'BLANK'
1110 2 18 SCAN END <-- NETWORKID: USIBMSY LUNAME: SYEC1DB2 LUWSEQ: 1
1111 2 16 INSERT --> 'BLANK'
1112 2 16 INSERT --> 'BLANK'
1113 2 233 CALL STORED 'BLANK'

LOCATION: STLEC1 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0)
GROUP: DSNCAT RECORD TRACE - SUMMARY REQUESTED FROM: ALL 17:38:00.00
MEMBER: V71A TO: DATES 17:40:00.00
SUBSYSTEM: V71A ACTUAL FROM: 01/30/10 17:38:00.83
DB2 VERSION: V10

ACE NUMBER ADDRESS
0
1 X'0583C8F8' 2 X'0583CE38'

RECORD TRACE COMPLETE

"ACE Cross-Reference Table" on page 37-3
For every trace specified, an ACE cross-reference table is printed for each location.

"Data Fields" on page 37-4
This topic describes the general format of the IFCID records presented in the summary record trace.
ACE Cross-Reference Table

For every trace specified, an ACE cross-reference table is printed for each location.

The columns of the ACE cross-reference table are:

**ACE NUMBER**
- The cross-reference number for the hexadecimal address of the agent control element. The lowest valid cross-reference number is 1. 0 indicates that the ACE address is not available.

**ACE ADDRESS**
- The hexadecimal address of the agent control element.
Data Fields

This topic describes the general format of the IFCID records presented in the summary record trace.

The records are presented in the requested sequence. There is one entry on the report for each record selected from the input data set, so the report can show more than one record of the same IFCID record type. Use the RECORD TIME field on the report to distinguish between records with the same IFCID record type.

Note:
1. An arrow (-->) pointing to the right on the trace indicates the beginning of an event.
2. An arrow (<--) pointing to the left on the trace indicates the end of an event.

Logical Unit of Work Identifiers on page 37-5
The logical unit of work identifiers are shown in the DATA column in front of the formatted data.

DDF Data on page 37-6
DDF data is only printed if there is a DDF header.

Accounting Token on page 37-7
All record trace reports show the value (in hexadecimal) of the accounting token in the DATA column when it contains a value other than blanks or binary zeros.
Logical Unit of Work Identifiers

The logical unit of work identifiers are shown in the DATA column in front of the formatted data.

NETWORKID: APCNET   LUNAME: SYDAPC4   LUWSEQ:   1

Field description

NETWORKID
The network ID.

LUNAME
The name of the logical unit.

LUWSEQ
The sequence number of the logical unit of work.
DDF Data

DDF data is only printed if there is a DDF header.

The following is printed in the DATA column after the formatted record:

REQUESTING LOCATION: USIBMSYSTDB2
REQUESTING TIMESTAMP: 01/30/10 18:54:53.90530718
AR NAME: USIBMSYSTDB2 PRDID: DB2 V9 M0

Field description

Here is a description of the field labels shown in the previous example:

REQUESTING LOCATION
The location requesting the work.

REQUESTING TIMESTAMP
The timestamp of the requester location.

AR NAME
The name of the application requester.

PRDID
The name, version, release, and modification level of the product making the request.
Accounting Token

All record trace reports show the value (in hexadecimal) of the accounting token in the DATA column when it contains a value other than blanks or binary zeros.

The Accounting token is used to correlate CICS records with DB2 records for the same task. If TOKENI=YES for TYPE=INIT, TOKENE=YES for TYPE=ENTRY, or both applies, in the resource control table, then the CICS logical unit of work ID (LUWID) minus the commit count (2 bytes) is passed to this field.

The first 8 bytes contain the network name, and the following 8 bytes contain the LU name. The final 6 bytes are the unique value.

ACCTKN X'00000000000000000000000000000000000000000000'
Chapter 38. The Short and Long Record Traces

The short and long record traces are similar. The short record trace reports non-serviceability data from records which are used by other subcomponents of the batch component. Serviceability records and fields are not printed on the short record trace. Only the occurrence of large records such as statistics, accounting, and system parameters is shown.

The long record trace reports all instrumentation facility records including Statistics, Accounting, and Performance records. The DB2 field names of serviceability fields are printed, as well as the occurrence of the serviceability records.

Depending on the record layout, the records are presented in either the DATA column or the full width of the report page.

If there is no data present for an IFCID, NO DATA is printed. If any unexpected data is found, it is printed in dump format. The dump format is also used for IFCID 0.

"The Short Record Trace" on page 38-2
The short record trace lists selected records from an input data set. It selects and formats nonserviceability data from the user-selected records that appear on other OMEGAMON XE for DB2 PE reports.

"The Long Record Trace" on page 38-3
The long record trace lists selected records from an input data set. It lists and formats all data from user-selected records.
The Short Record Trace

The short record trace lists selected records from an input data set. It selects and formats nonserviceability data from the user-selected records that appear on other OMEGAMON XE for DB2 PE reports.

Use the short record trace to access the DB2 nonserviceability data used by OMEGAMON XE for DB2 PE, and to access data not presented in other reports.

Some long records (for example, system statistics) are ignored by the short record trace and some records are shown in abbreviated form.

The short record trace can be used with all the selection options such as INCLUDE and EXCLUDE.

The following command produces the short record trace example shown in "Example of the Short Record Trace."

```
RECTRACE
TRACE
FROM (,21:54:00)
TO (,21:56:00)
```

Example of the Short Record Trace

This is an example of a short record trace produced by the previous command:

```
LOCATION: PMODE1 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-1
GROUP: DBE1 RECORD TRACE - SHORT REQUESTED FROM: ALL 21:54:00.00
MEMBER: SE11 TO: DATES 21:56:00.00
SUBSYSTEM: SE11 ACTUAL FROM: 07/15/13 21:54:00.11
DB2 VERSION: V11 PAGE DATE: 07/15/13
PRIMAUTH CONNECT INSTANCE END_USER WS_NAME TRANSACT
ORIGAUTH CORRNAME CONNTYPE RECORD TIME DESTNO ACE IFC DESCRIPTION DATA
-------- -------- ----------- ----------------- ------ --- --- -------------- ------------------------------------------------------
N/P N/P CBA37B3FD9B8 N/P N/P N/P N/P N/P N/P 'BLANK' 21:54:00.11351049 91109 1 1 SYSTEM STATS NETWORKID: SE11 LUNAME: SE11 LUWSEQ: 1
N/P N/P N/P 'BLANK' 21:54:03.28092609 91110 1 2 DB STATISTICS NETWORKID: SE11 LUNAME: SE11 LUWSEQ: 1
SYSOPS SE11 CBA37B3FD9B8 N/P N/P N/P N/P N/P N/P N/P 'BLANK' 21:54:03.28111366 91111 1 106 SYS PARAMETERS NETWORKID: SE11 LUNAME: SE11 LUWSEQ: 1
N/P N/P N/P 'BLANK' 21:55:00.30227550 91121 1 1 SYSTEM STATS NETWORKID: SE11 LUNAME: SE11 LUWSEQ: 1
N/P N/P N/P 'BLANK' 21:55:00.34404053 91122 1 2 DB STATISTICS NETWORKID: SE11 LUNAME: SE11 LUWSEQ: 1
```

RECORD TRACE COMPLETE
The Long Record Trace

The long record trace lists selected records from an input data set. It lists and formats all data from user-selected records.

Use the long record trace to produce a formatted report of all data in the selected trace records.

The long record trace can be used with all the selection options such as INCLUDE and EXCLUDE.

Note: A long record trace can show a great amount of data. Consider limiting the size of the report with the INCLUDE, EXCLUDE, FROM, and TO options of the TRACE subcommand.

The following command produces the long record trace example in "Example of a Long Record Trace."

```
RECTRACE
| TRACE
| FROM ,21:54:00)
| TO ,21:56:00)
| LEVEL(LONG)
```

Example of a Long Record Trace

This is an example of a long record trace produced by the previous command:

```
LOCATION: PMODEBE1  OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0)
GROUP: DBE1  RECORD TRACE - LONG
MEMBER: SE11  FROM ALL 21:54:00.00
SUBSYSTEM: SE11  TO DATES 21:56:00.00
DB2 VERSION: V11  ACTUAL FROM: 07/15/13 21:54:00.11

PRIMAUTH CONNECT INSTANCe END_USER WS_NAME IFAC DESCRIPTION DATA
ORIGAUTH CORRNAME CONNTYPE RECORD TIME DESTNO ACE
-------- -------- ----------- ----------------- ------ --- --- -------------- ------------------------------------------------------
N/P N/P CBA37B3FD9B8 N/P N/P N/P
N/P N/P 'BLANK' 21:54:00.11351049 91109 1 1 SYSTEM STATS NETWORKID: SE11 LUNAME: SE11 LUWSEQ: 1
N/P N/P N/P

DESTINATION RELATED DATA
| DEST NAME | SMF | SEQNO | 91108 | RECS WRITTEN | 91108 | RECS NOT WRITTEN | 0 BUFFER ERRORS | 0 |
| QWSBOTH3 | 0 | QWSBOTH4 | 0 |

| DEST NAME | RES | SEQNO | 0 | RECS WRITTEN | 0 | RECS NOT WRITTEN | 0 BUFFER ERRORS | 0 |
| QWSBOTH3 | 0 | QWSBOTH4 | 0 |

| DEST NAME | GTF | SEQNO | 0 | RECS WRITTEN | 0 | RECS NOT WRITTEN | 0 BUFFER ERRORS | 0 |
| QWSBOTH3 | 0 | QWSBOTH4 | 0 |

| DEST NAME | SRV | SEQNO | 0 | RECS WRITTEN | 0 | RECS NOT WRITTEN | 0 BUFFER ERRORS | 0 |
| QWSBOTH3 | 0 | QWSBOTH4 | 0 |

| DEST NAME | SR1 | SEQNO | 516259 | RECS WRITTEN | 99 | RECS NOT WRITTEN | 99 BUFFER ERRORS | 0 |
| QWSBOTH3 | 0 | QWSBOTH4 | 0 |
```
## Record Trace

<table>
<thead>
<tr>
<th>IFCID</th>
<th>SID</th>
<th>SEQNO</th>
<th>RECS WRITTEN</th>
<th>RECS NOT WRITTEN</th>
<th>RECS NOT DESIRED</th>
<th>BUFFER NOT AVAILABLE</th>
<th>COLLECT FAILURES</th>
<th>DATA</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td></td>
<td>766</td>
<td>12550</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>3159</td>
<td>1026</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>73</td>
<td>73</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>65</td>
<td>65</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>106</td>
<td></td>
<td>7971</td>
<td>12607</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>140</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>141</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>142</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>143</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>144</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>145</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

Location: PMODEB1
OMEGAMON XE for DB2 Performance Expert (V5R2M0)

Group: DBE1
DB2 Trace - Long
Requested from: ALL 21:54:00.00
Actual from: 07/15/13 21:54:00.11

Member: SE11
To: Dates 21:56:00.00

Subsystem: SE11
Actual from: 07/15/13 21:54:00.11
Page date: 07/15/13
Chapter 38. The Short and Long Record Traces
### Record Trace

**DDF Data by Location**

<table>
<thead>
<tr>
<th>Location Name (Short)</th>
<th>Location Name (Long)</th>
<th>Initiated Conversations</th>
<th>Deallocated Conversations</th>
<th>Messages Sent to Remote</th>
<th>SQL STMS Sent to Remote</th>
<th>Bytes Sent to Remote</th>
<th>Blocks Transmitted</th>
<th>N/P</th>
<th>N/P</th>
<th>N/P</th>
<th>N/P</th>
<th>N/P</th>
<th>N/P</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRDA Remote Locs</td>
<td>DRDA Remote Locs</td>
<td>0</td>
<td>0</td>
<td>12287</td>
<td>0</td>
<td>402547183</td>
<td>12171</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Location:** PMODE1

**OMEGAMON XE for DB2 Performance Expert (V5R2M0)**

**Page:** 1-6

**Group:** DBE1

**Member:** SE11

**Subsystem:** SE11

**Page Date:** 07/15/13 21:54:00.11

**DB2 Version:** V11

**SUBSYSTEM: SE11**

**LOCATION NAME (SHORT):** DRDA Remote Locs

**LOCATION NAME (LONG):** DRDA Remote Locs

**Initiated Conversations:** 0

**Deallocated Conversations:** 0

**Messages Sent to Remote:** 12287

**SQL STMS Sent to Remote:** 0

**Bytes Sent to Remote:** 402547183

**Blocks Transmitted:** 12171

**PRIMARY CONNECT INSTANCE END USER WS_NAME TRANSACT**

<table>
<thead>
<tr>
<th>Original Auth</th>
<th>CorrName</th>
<th>Conntype</th>
<th>Record Time</th>
<th>DestName</th>
<th>ifc</th>
<th>Description</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRIMAUTH</td>
<td>CORRNAME</td>
<td>CONNTYPE</td>
<td>RECORD TIME</td>
<td>DESTNO</td>
<td>ACE</td>
<td>IFC</td>
<td>DATA</td>
</tr>
<tr>
<td>DB2 VERSION: V11 PAGE: 1-6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**GLOBAL DDF DATA**

<table>
<thead>
<tr>
<th>IBAT/CONNECT MAX ACTIVE</th>
<th>IBAT CONNECTED MAX CONNECTED</th>
<th>0</th>
</tr>
</thead>
</table>
| IBAT CONNECTED MAX CONNECTED | 0 |}

**Location:** PMODE1

**OMEGAMON XE for DB2 Performance Expert (V5R2M0)**

**Page:** 1-7

**Group:** DBE1

**Member:** SE11

**Subsystem:** SE11

**Page Date:** 07/15/13 21:54:00.11

**DB2 Version:** V11

**SUBSYSTEM: SE11**

**LOCATION NAME (SHORT):** DRDA Remote Locs

**LOCATION NAME (LONG):** DRDA Remote Locs

**Initiated Conversations:** 0

**Deallocated Conversations:** 0

**Messages Sent to Remote:** 12287

**SQL STMS Sent to Remote:** 0

**Bytes Sent to Remote:** 402547183

**Blocks Transmitted:** 12171

**PRIMARY CONNECT INSTANCE END USER WS_NAME TRANSACT**

<table>
<thead>
<tr>
<th>Original Auth</th>
<th>CorrName</th>
<th>Conntype</th>
<th>Record Time</th>
<th>DestName</th>
<th>ifc</th>
<th>Description</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRIMAUTH</td>
<td>CORRNAME</td>
<td>CONNTYPE</td>
<td>RECORD TIME</td>
<td>DESTNO</td>
<td>ACE</td>
<td>IFC</td>
<td>DATA</td>
</tr>
<tr>
<td>DB2 VERSION: V11 PAGE: 1-7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**GLOBAL DDF DATA**

<table>
<thead>
<tr>
<th>IBAT/CONNECT MAX ACTIVE</th>
<th>IBAT CONNECTED MAX CONNECTED</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBAT CONNECTED MAX CONNECTED</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
Record Trace

---

<table>
<thead>
<tr>
<th>SQL CALL DATA</th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECT ..........</td>
</tr>
<tr>
<td>DELETE ..........</td>
</tr>
<tr>
<td>OPEN ..........</td>
</tr>
<tr>
<td>COMMENT ON ........</td>
</tr>
<tr>
<td>REVOKE ..........</td>
</tr>
<tr>
<td>DESCRIBE TABLE ...</td>
</tr>
<tr>
<td>RELEASE ..........</td>
</tr>
<tr>
<td>RENAME TABLE ......</td>
</tr>
<tr>
<td>MERGE ..........</td>
</tr>
<tr>
<td>CREATE DATABASE ..</td>
</tr>
<tr>
<td>CREATE STOGROUP ..</td>
</tr>
<tr>
<td>CREATE TABLE .....</td>
</tr>
<tr>
<td>CREATE INDEX .....</td>
</tr>
<tr>
<td>CREATE VIEW .....</td>
</tr>
<tr>
<td>CREATE SYNONYM ...</td>
</tr>
<tr>
<td>CREATE ALIAS .....</td>
</tr>
<tr>
<td>CREATE TRIGGER ...</td>
</tr>
<tr>
<td>CREATE DIST TYPE :</td>
</tr>
<tr>
<td>CREATE PROCEDURE :</td>
</tr>
<tr>
<td>CREATE FUNCTION :</td>
</tr>
<tr>
<td>ROLLBACK .......</td>
</tr>
<tr>
<td>CREATE CURSOR CONT:</td>
</tr>
<tr>
<td>CREATE MASK/PERM :</td>
</tr>
<tr>
<td>CREATE VARIABLE ...</td>
</tr>
<tr>
<td>ISET CUR SQL ID ...</td>
</tr>
<tr>
<td>ISET CUR DEGREE ...</td>
</tr>
<tr>
<td>ISET CUR PRECISION:</td>
</tr>
<tr>
<td>MULTI-ROW PROCESSING:</td>
</tr>
<tr>
<td>ROWS Fetched ....</td>
</tr>
<tr>
<td>ROWS Inserted ....</td>
</tr>
<tr>
<td>ROWS Deleted ....</td>
</tr>
</tbody>
</table>

---

Chapter 38. The Short and Long Record Traces 38-7
### Record Trace

<table>
<thead>
<tr>
<th>PLANNAME</th>
<th>CORRNMBR</th>
<th>TCB CPU TIME</th>
<th>ID</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Query Parallelism

- MAX DEG ESTIMATED: 8
- MEMBERS SKIPPED: 0
- PARALL. DISABLED: N/A
- MAX DEG PLANNED: 8
- MAX DEG EXECUTED: 8
- PARALLEL GROUPS:
  - IFALL TO SEQ-CURSOR: 0
  - IFALL TO SEQ-NOSAME: 0
  - IFALL TO SEQ-STORE: 0
  - IFALL TO SEQ-NEGOTN: 0
  - IFALL TO SEQ-A-PROC: 0
  - IFALL TO SEQ-ENCLY: N/A

#### Dynamic SQL Stmt

- BIND PLAN (REPL): 3
- BIND PACK (REPL): 3
- DS CLOSED-THRESH.REACH: 0
- PKG-AUTH.SUCC-PUBL: 6
- PLANS BOUND: 3
- PACKAGES BOUND: 3
- DS NOUSE,NOCLOSE-HWM: 104
- AUTH.SUCC-NO CAT.: 2528
- PLAN ALLOC.ATTMP: 2560
- PACK.ALLOC.ATTMP: 248
- OPEN DATASETS - CURR.: 104
- AUTHORIZ.SUCCESS.5239
- PLAN ALLOC.ATTMP: 2560
- PACK.ALLOC.ATTMP: 248
- OPEN DATASETS - HWM: 104
- AUTHORIZ.ATTEMPTS: 5239
- PLAN EXEC.ATTMP: 1
- PACK.EXEC.ATTMP: 0
- PKG-AUTH.SUCC-PUB: 6
- DS NOUSE,NOCLOSE-CURR.: 104
- AUTH.SUCC-PUBLIC: 25
- PLAN ALLOC.ATTMP: 2560
- PACK.ALLOC.ATTMP: 248
- OPEN DATASETS - CURR.: 104
- AUTHORIZ.SUCCESS.5239
- PLAN EXEC.ATTMP: 1
- PACK.EXEC.ATTMP: 0
- PKG-AUTH.SUCC-PUB: 6
- DS NOUSE,NOCLOSE-HWM: 104
- AUTH.SUCC-NO CAT.: 2528

#### Service Controller Data

- PLAN ALLOC.ATTMP: 2560
- PACK.ALLOC.ATTMP: 248
- OPEN DATASETS - HWM: 104
- AUTHORIZ.ATTEMPTS: 5239
- PLAN ALLOC.SUCC.: 2560
- PACK.ALLOC.SUCC.: 247
- OPEN DATASETS - CURR.: 104
- AUTHORIZ.SUCCESS.: 5239
- PLANS BOUND: 3
- PACKAGES BOUND: 3
- DS MOUNT,NOCLOSE-HWM: 104
- AUTH.SUCC-PUBLIC: 25
- BIND PLAN (ADD): 0
- BIND PACK (ADD): 0
- DS NOUSE,NOCLOSE-CURR.: 104
- AUTH.SUCC-PUBLIC: 25
- BIND PLAN (REPL): 3
- BIND PACK (REPL): 3
- DS CLOSED-THRESH.REACH: 0
- PKG-AUTH.SUCC-PUBL: 16
- AUTOBILL.PLAN ATTMP: 1
- AUTOBILL.PACK ATTMP: 0
- R/W TO R/O CONVERSIONS: 1628
- PKG-AUTH.SUCC-PUBL: 6

---

**Note:** The above content appears to be a part of a database performance monitoring report, showing various statistics and parameters related to SQL statement execution, database operations, and system performance metrics. The records include details on plan execution, parallel query processing, dynamic SQL statement processing, service controller data, and miscellaneous system statistics. The data is presented in a tabular format with columns for different data categories such as plan name, correlation number, TCB CPU time, and ID.
### DB2 Version 10 or Higher Section:

- Total Fract Storage (KB): N/A
- Total Whole 4K Storage (MB): N/A
- Total Whole 32K Storage (MB): N/A

### DB2 Version 9 Section:

- Total 4K Storage Used (KB): 0
- Total 32K Storage Used (KB): 0
- Total Whole 4K Storage (MB): N/A
- Total Whole 32K Storage (MB): N/A

### Data Manager Data

- CUR RIDLIST Blocks .................: 0
- CUR RIDLIST BlocksOverflowed ......: 0
- RIDLIST Terminated-DS Limit ........: 0
- RIDLIST Terminated-DS Limit ......: 0
- RIDLIST Terminated-No Storage ....: 0
- RIDLIST Terminated-No Storage ...: 0

#### LOCATION:

- PMODEBE1

#### DB2 PERFORMANCE EXPERT (V5R2M0)

- REQUESTED FROM: ALL
- PAGE: 07/15/13
- PAGE DATE: 07/15/13

#### Record Trace

- Record Trace

#### Chapter 38. The Short and Long Record Traces

- Chapter 38. The Short and Long Record Traces
### Record Trace

**DB2 Version:** VII  
**Page Date:** 07/15/13

<table>
<thead>
<tr>
<th>Column</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plan Name</td>
<td>DB2 PLANATB STORAGE : 29568 PKG ATB STORAGE : 0</td>
</tr>
<tr>
<td>Request Time</td>
<td>21:54:03.28092609</td>
</tr>
<tr>
<td>Resource ID</td>
<td>91110</td>
</tr>
<tr>
<td>DB Statistics</td>
<td>1 2</td>
</tr>
</tbody>
</table>

**Database Statistics**

- **CPU Time:** 75.80329233
- **SYSOPR:** 016.WVSM
- **NETWORKID:** SE11
- **LUWSEQ:** 1
- **SYSTEM Initialization Parameters**

**System Parameters**

- **User XML Values (KB):** 204800
- **System XML Values (MB):** 10240
- **MAX OPEN DS FOR LOB:** 100
- **LOB INLINE LENGTH:** 0
- **COMPRESS:** OFF
- **RANDOMIZE XML DOCID:** NO

**Other Parameters**

- **User Lob Value Store:** 10240
- **User Lob Value Store:** 4096
- **System XML Values (MB):** 10240

---

**Performance Expert (VSR2MO)**

**Page:** 1-15  
**GROUP:** DB21

**DB Version:** VII  
**Page Date:** 07/15/13

<table>
<thead>
<tr>
<th>Column</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plan Name</td>
<td>GROUP BUFFER POOL ID ............</td>
</tr>
<tr>
<td>Request Time</td>
<td>21:54:03.28092609</td>
</tr>
<tr>
<td>Resource ID</td>
<td>91110</td>
</tr>
<tr>
<td>DB Statistics</td>
<td>1 2</td>
</tr>
</tbody>
</table>

**Database Statistics**

- **CPU Time:** 75.80329233
- **SYSOPR:** 016.WVSM
- **NETWORKID:** SE11
- **LUWSEQ:** 1
- **SYSTEM Initialization Parameters**

**System Parameters**

- **User XML Values (KB):** 204800
- **System XML Values (MB):** 10240
- **MAX OPEN DS FOR LOB:** 100
- **LOB INLINE LENGTH:** 0
- **COMPRESS:** OFF
- **RANDOMIZE XML DOCID:** NO

**Other Parameters**

- **User Lob Value Store:** 10240
- **User Lob Value Store:** 4096
- **System XML Values (MB):** 10240

---

**Performance Expert (VSR2MO)**

**Page:** 1-18  
**GROUP:** DB21

**DB Version:** VII  
**Page Date:** 07/15/13

<table>
<thead>
<tr>
<th>Column</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plan Name</td>
<td>GROUP BUFFER POOL ID ............</td>
</tr>
<tr>
<td>Request Time</td>
<td>21:54:03.28092609</td>
</tr>
<tr>
<td>Resource ID</td>
<td>91110</td>
</tr>
<tr>
<td>DB Statistics</td>
<td>1 2</td>
</tr>
</tbody>
</table>

**Database Statistics**

- **CPU Time:** 75.80329233
- **SYSOPR:** 016.WVSM
- **NETWORKID:** SE11
- **LUWSEQ:** 1
- **SYSTEM Initialization Parameters**

**System Parameters**

- **User XML Values (KB):** 204800
- **System XML Values (MB):** 10240
- **MAX OPEN DS FOR LOB:** 100
- **LOB INLINE LENGTH:** 0
- **COMPRESS:** OFF
- **RANDOMIZE XML DOCID:** NO

**Other Parameters**

- **User Lob Value Store:** 10240
- **User Lob Value Store:** 4096
- **System XML Values (MB):** 10240
Chapter 38. The Short and Long Record Traces

Record Trace
Record Trace

| LOCATION: PMODEBE | OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (YRS2M0) | PAGE: 1-20 |
| GROUP: DBE1 | RECORD TRACE - LONG | REQUESTED FROM: 21:54:00.00 |
| SUBSYSTEM: SE11 | TO: DATES 21:56:00.00 |
| DB2 VERSION: V11 | ACTUAL FROM: 07/15/13 21:54:00.11 |
| SYSPRO SE11 CBAA7B6F9DBB N/P N/P |
| SYSPRO 016.WVSM 'BLANK' 21:54:03.28113166 91111 1 106 SYS PARAMETERS |

**'BLANK' T 01**

---

### Subsystem

- **Subsystem:** SE11
- **Actual From:** 07/15/13 21:54:00.11

### Location

- **Location:** PMODEBE
- **OMEGAMON XE for DB2 Performance Expert (YRS2M0)**
- **Page:** 1-20

### Member

- **Member:** SE11
- **Requested From:** 21:54:00.00

### Group

- **Group:** DBE1
- **To:** Dates 21:56:00.00

### Subsystem Version

- **Subsystem Version:** V11
- **Actual From:** 07/15/13 21:54:00.11

### Syspro

- **Syspro:** SE11 CBAA7B6F9DBB N/P N/P
- **Syspro 016.WVSM 'BLANK' 21:54:03.28113166 91111 1 106 SYS PARAMETERS

### Record Trace - Long Requested From All

<table>
<thead>
<tr>
<th>PRIMAUTH</th>
<th>CORNAME</th>
<th>CONTYPE</th>
<th>RECORD TIME</th>
<th>DESTNO</th>
<th>ACE</th>
<th>IFC</th>
<th>DESCRIPTION</th>
<th>DATA</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYSOPR</td>
<td>SE11</td>
<td>CBA37B3FD9B8</td>
<td>N/P</td>
<td>N/P</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Planname

- **Planname:** PMODEBE
- **OMEGAMON XE for DB2 Performance Expert (YRS2M0)**
- **Page:** 1-20

### Member

- **Member:** SE11
- **Requested From:** 21:54:00.00

---

### Records Trace

1. **Record Trace**
   - **SYSOPR 016.WVSM 'BLANK' 21:54:03.28113166 91111 1 106 SYS PARAMETERS**

2. **Planname**
   - **PMODEBE 21:54:03.28113166 91111 1 106 SYS PARAMETERS**

---

### System Parameters

- **System Parameters**
  - **DB2 Version:** V11
  - **Page Date:** 07/15/13

---

### System Member

- **System Member:** SE11
- **OMEGAMON XE for DB2 Performance Expert (V5R2M0)**
- **Page:** 1-21

### System Group

- **System Group:** DBE1
- **OMEGAMON XE for DB2 Performance Expert (V5R2M0)**
- **Page:** 1-21

---

### System Location

- **System Location:** PMODEBE
- **OMEGAMON XE for DB2 Performance Expert (YRS2M0)**
- **Page:** 1-21

### System Member

- **System Member:** SE11
- **OMEGAMON XE for DB2 Performance Expert (YRS2M0)**
- **Page:** 1-21

---

### System Group

- **System Group:** DBE1
- **OMEGAMON XE for DB2 Performance Expert (YRS2M0)**
- **Page:** 1-21

---

### System Location

- **System Location:** PMODEBE
- **OMEGAMON XE for DB2 Performance Expert (YRS2M0)**
- **Page:** 1-21

---

### System Member

- **System Member:** SE11
- **OMEGAMON XE for DB2 Performance Expert (YRS2M0)**
- **Page:** 1-21

---

### System Group

- **System Group:** DBE1
- **OMEGAMON XE for DB2 Performance Expert (YRS2M0)**
- **Page:** 1-21

---

### System Location

- **System Location:** PMODEBE
- **OMEGAMON XE for DB2 Performance Expert (YRS2M0)**
- **Page:** 1-21

---

### System Member

- **System Member:** SE11
- **OMEGAMON XE for DB2 Performance Expert (YRS2M0)**
- **Page:** 1-21

---

### System Group

- **System Group:** DBE1
- **OMEGAMON XE for DB2 Performance Expert (YRS2M0)**
- **Page:** 1-21

---

### System Location

- **System Location:** PMODEBE
- **OMEGAMON XE for DB2 Performance Expert (YRS2M0)**
- **Page:** 1-21

---

### System Member

- **System Member:** SE11
- **OMEGAMON XE for DB2 Performance Expert (YRS2M0)**
- **Page:** 1-21

---

### System Group

- **System Group:** DBE1
- **OMEGAMON XE for DB2 Performance Expert (YRS2M0)**
- **Page:** 1-21

---

### System Location

- **System Location:** PMODEBE
- **OMEGAMON XE for DB2 Performance Expert (YRS2M0)**
- **Page:** 1-21

---

### System Member

- **System Member:** SE11
- **OMEGAMON XE for DB2 Performance Expert (YRS2M0)**
- **Page:** 1-21

---

### System Group

- **System Group:** DBE1
- **OMEGAMON XE for DB2 Performance Expert (YRS2M0)**
- **Page:** 1-21

---

### System Location

- **System Location:** PMODEBE
- **OMEGAMON XE for DB2 Performance Expert (YRS2M0)**
- **Page:** 1-21

---

### System Member

- **System Member:** SE11
- **OMEGAMON XE for DB2 Performance Expert (YRS2M0)**
- **Page:** 1-21

---

### System Group

- **System Group:** DBE1
- **OMEGAMON XE for DB2 Performance Expert (YRS2M0)**
- **Page:** 1-21

---

### System Location

- **System Location:** PMODEBE
- **OMEGAMON XE for DB2 Performance Expert (YRS2M0)**
- **Page:** 1-21

---

### System Member

- **System Member:** SE11
- **OMEGAMON XE for DB2 Performance Expert (YRS2M0)**
- **Page:** 1-21

---

### System Group

- **System Group:** DBE1
- **OMEGAMON XE for DB2 Performance Expert (YRS2M0)**
- **Page:** 1-21
Chapter 38. The Short and Long Record Traces

Record Trace

...
Record Trace

<table>
<thead>
<tr>
<th>PRIMAUTH</th>
<th>CORRNAME</th>
<th>CONNTYPE</th>
<th>RECORD TIME</th>
<th>DESTNO</th>
<th>ACE</th>
<th>IFC</th>
<th>DESCRIPTION</th>
<th>DATA</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYSOPR</td>
<td>SE11</td>
<td>'BLANK'</td>
<td>21:54:03.84</td>
<td>91118</td>
<td>2</td>
<td>105</td>
<td>STORAGEMGR</td>
<td>0.00010481</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>N/P</td>
<td>N/P</td>
<td>CBA442E7D4D</td>
<td>N/P</td>
<td>N/P</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>'BLANK'</td>
<td>AT01</td>
<td>'BLANK'</td>
<td>21:54:07.91</td>
<td>91118</td>
<td>2</td>
<td>225</td>
<td>STORAGE MGR</td>
<td>512000</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>N/P</td>
<td>N/P</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

...
### ADDRESS SPACE SUMMARY - DIST

- **24-BIT LOW PRIVATE**: 249856
- **24-BIT HIGH PRIVATE**: 270336
- **31-BIT EXTENDED LOW PRIVATE**: 6397952
- **31-BIT EXTENDED HIGH PRIVATE**: 14110720
- **CURR HIGH ADDR 24-BIT PRIV REGION**: X'00043000'
- **CURR HIGH ADDR 31-BIT PRIV REGION**: X'2121A000'
- **31-BIT RESERVED FOR MVS**: 26040960
- **STORAGE CUSHION WARNING TO CONTRACT**: 159802982
- **TOTAL 31-BIT GETMAINED STACK**: 1212416
- **TOTAL 31-BIT STACK IN USE**: 933888
- **TOTAL 31-BIT VARIABLE POOL**: 286720
- **TOTAL 31-BIT FIXED POOL**: 106496
- **TOTAL 31-BIT GETMAINED**: 4464
- **AMOUNT OF AVAILABLE 31-BIT**: 1577517056
- **SYSTEM AGENT STACK STORAGE IN USE**: 802816

### 64-BIT VARIABLE POOL

- **TOTAL 64-BIT VARIABLE POOL**: 36864
- **TOTAL 64-BIT FIXED**: 98304
- **TOTAL 64-BIT GETMAINED**: 0
- **TOTAL 64-BIT PRIVATE FOR STOR MANAG**: 1400832

### THREAD INFORMATION

- **ACTIVE THREADS**: 10
- **ACTIVE AND DISCONNECTED DBATS**: 0
- **CASTOUT ENGINES**: 34
- **DEFERRED WRITE ENGINES**: 0
- **GBP WRITE ENGINES**: 1
- **PREFetch ENGINES**: 7
- **P-LOCK/NOTIFY EXIT ENGINES**: 4
- **PARALLEL CHILD THREADS**: 0

### SHARED/COMMON STORAGE SUMMARY

- **31-BIT COMMON FIXED POOL STORAGE**: 1036288
- **31-BIT COMMON VARIABLE POOL STORAGE**: 69320
- **31-BIT COMMON GETMAINED STORAGE**: 31640172
- **EXTENDED CSA SIZE**: 314601472
- **64-BIT COMMON FIXED POOL STORAGE**: 5767168
- **64-BIT COMMON VARIABLE POOL STORAGE**: 203460688
- **64-BIT COMMON GETMAINED STORAGE**: 1400832
- **64-BIT SHARED VARIABLE POOL STORAGE**: 21325448
- **64-BIT SHARED FIXED POOL STORAGE**: 3559424
- **64-BIT SHARED GETMAINED STORAGE**: 148582360
- **64-BIT SHARED STORAGE-STOR MGR CTRL**: 1400832
- **64-BIT SHARED SYSTEM AGENT STACK (AS)**: 3879312
- **64-BIT SHARED NON-SYSTEM AS**: 5428800
- **SHARED MEMORY OBJECTS**: 11
- **64-BIT SHARED MEMORY PAGES**: 721420288
- **WMM FOR 64-BIT SHARED BYTES**: 29543799648
- **64-BIT SHARED PAGES BACKED IN REAL**: 13442
- **AUX SLOTS USED FOR 64-BIT SHARED STOR**: 56295
- **64-BIT PAGES PAGED IN FROM AUX STOR**: 145124
- **64-BIT SHARED STG REAL 4K FRMS IN USE**: 397
- **64-BIT STACK STG REAL 4K FRMS IN USE**: 65
- **64-BIT COMMON STG REAL 4K FRMS IN USE**: 89
- **LOG MGR WRITE BUFFER FRAMES IN REAL**: 1004
- **LOG MGR WRITE BUFFER FRAMES IN AUX**: 30
- **QW0225_WARN**
  - 1
- **QW0225_REALAVAIL**
  - 1244
- **QW0225_REALAVAILD**
  - 1188
- **QW0225_ESDA_HMM**
  - 146554880
- **QW0225_ESDA_HMM**
  - 24973688
- **QW0225_ECSA alloc**
  - 23918144
- **QW0225_ECSA alloc**
  - 141291504
- **QW0225_ECSA alloc**
  - 0
### Record Trace

<table>
<thead>
<tr>
<th>PRIMAUTH</th>
<th>CONNECT</th>
<th>INSTANCE</th>
<th>END_USER</th>
<th>WS_NAME</th>
<th>TRANSACT</th>
<th>ORIGAUTH</th>
<th>CORRNAME</th>
<th>CONNTYPE</th>
<th>RECORD</th>
<th>TIME</th>
<th>DESTNO</th>
<th>ACE</th>
<th>IFC</th>
<th>DESCRIPTION</th>
<th>DATA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| N/P | N/P | N/P | 'BLANK' | 21:54:07.81536325 | 91120 | 1 | 225 | STORAGE MGR |
|-----|-----|-----|---------|------------------|-------|---|-----|------------|--------|
| 0   | 0   | 0   | 0       |                  |       |   |     |           |        |

#### EXECUTION CONTEXT STACK

**STATEMENT CACHE / xPROCS**:
- Allocated stor for dyn sql stmts: 106496
- Allocated stor for static sql stmts: 212992
- Total 31-bit xproc dynamic sql: N/A
- Total 31-bit xproc static sql: N/A
- Statements in 64-bit agent local pools (ALP): 0
- Allocated stmt cache in 64-bit ALP: 0
- Timestamp of HWM after last 225 REC: 07/15/13 21:53:00.433538
- Total 64-bit stmt cache blks 2G: 167936

**STORAGE POOL DETAILS**
- 31-BIT DBM1 PRIVATE VARIABLE POOLS:
  - Agent local storage: 516096
  - System agent storage: 339968
  - Buffer manager storage blocks: 610304
- 64-BIT POOLS:
  - Shared agent local (variable pool): 1964416
  - IRB pool storage (fixed pool): 0
  - Array variable storage: 0

**IRLM POOL STATISTICS**
- Above the bar values:
  - ATB CSA CURRENT: 0
  - ATB CSA HIGH WATER MARK: 0
  - ATB PRIVATE CURRENT: 0
  - ATB PRIVATE HIGH WATER MARK: 0
  - ATB PRIVATE MAX AVAILABILITY: 0
- Below the bar values:
  - BTB PRIVATE CURRENT: 0
  - BTB PRIVATE HIGH WATER MARK: 0
  - BTB PRIVATE MAX AVAILABILITY: 0
- ECSA:
  - ECSA CURRENT: 2036775
  - ECSA HIGH WATER MARK: 2081831
Record Trace

DESTINATION RELATED DATA

DEST NAME SMF SEQUO 91120 RECS WRITTEN 91120 RECS NOT WRITTEN 0 BUFFER ERRORS 0
<table>
<thead>
<tr>
<th>QWSBOTH1</th>
<th>QWSBOTH2</th>
<th>QWSBOTH3</th>
<th>QWSBOTH4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

DEST NAME RES SEQUO 0 RECS WRITTEN 0 RECS NOT WRITTEN 0 BUFFER ERRORS 0
<table>
<thead>
<tr>
<th>QWSBOTH1</th>
<th>QWSBOTH2</th>
<th>QWSBOTH3</th>
<th>QWSBOTH4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

DEST NAME GTF SEQUO 0 RECS WRITTEN 0 RECS NOT WRITTEN 0 BUFFER ERRORS 0
<table>
<thead>
<tr>
<th>QWSBOTH1</th>
<th>QWSBOTH2</th>
<th>QWSBOTH3</th>
<th>QWSBOTH4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

DEST NAME SRV SEQUO 0 RECS WRITTEN 0 RECS NOT WRITTEN 0 BUFFER ERRORS 0
<table>
<thead>
<tr>
<th>QWSBOTH1</th>
<th>QWSBOTH2</th>
<th>QWSBOTH3</th>
<th>QWSBOTH4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

DEST NAME SR1 SEQUO 0 RECS WRITTEN 516269 RECS NOT WRITTEN 99 BUFFER ERRORS 0
<table>
<thead>
<tr>
<th>QWSBOTH1</th>
<th>QWSBOTH2</th>
<th>QWSBOTH3</th>
<th>QWSBOTH4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>99</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

DEST NAME SR2 SEQUO 24128 RECS WRITTEN 24128 RECS NOT WRITTEN 0 BUFFER ERRORS 0
<table>
<thead>
<tr>
<th>QWSBOTH1</th>
<th>QWSBOTH2</th>
<th>QWSBOTH3</th>
<th>QWSBOTH4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

DEST NAME OP1 SEQUO 45 RECS WRITTEN 45 RECS NOT WRITTEN 0 BUFFER ERRORS 0
<table>
<thead>
<tr>
<th>QWSBOTH1</th>
<th>QWSBOTH2</th>
<th>QWSBOTH3</th>
<th>QWSBOTH4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

RECORD TRACE COMPLETE

Chapter 38. The Short and Long Record Traces 38-17
Accounting Report
# Chapter 39. Dump Record Trace

The dump record trace lists all data from selected records of an input data set.

The following command produces the dump record trace example shown in "Dump Record Trace Example."

```
RECTRACE
TRACE LEVEL (DUMP)
```

## Dump Record Trace Example

Here is an example of a dump record trace.

<table>
<thead>
<tr>
<th>Address</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
<td>08500000</td>
</tr>
<tr>
<td>0020</td>
<td>C4F8F1F1</td>
</tr>
<tr>
<td>0040</td>
<td>B058F1A0</td>
</tr>
<tr>
<td>0060</td>
<td>00000040</td>
</tr>
<tr>
<td>0080</td>
<td>00001024</td>
</tr>
<tr>
<td>0100</td>
<td>00000853</td>
</tr>
<tr>
<td>0120</td>
<td>00000000</td>
</tr>
<tr>
<td>0140</td>
<td>00000000</td>
</tr>
<tr>
<td>0160</td>
<td>00000000</td>
</tr>
<tr>
<td>0180</td>
<td>00000000</td>
</tr>
<tr>
<td>0200</td>
<td>00000000</td>
</tr>
<tr>
<td>0220</td>
<td>00000000</td>
</tr>
<tr>
<td>0240</td>
<td>00000000</td>
</tr>
<tr>
<td>0260</td>
<td>00000000</td>
</tr>
<tr>
<td>0280</td>
<td>00000000</td>
</tr>
<tr>
<td>0300</td>
<td>00000000</td>
</tr>
<tr>
<td>0320</td>
<td>00000000</td>
</tr>
<tr>
<td>0340</td>
<td>00000000</td>
</tr>
<tr>
<td>0360</td>
<td>00000000</td>
</tr>
<tr>
<td>0380</td>
<td>00000000</td>
</tr>
</tbody>
</table>

**Note:**

- The dump record trace shows the hexadecimal representation of memory addresses and data values.
- The example includes addresses and data values that are part of a dump record trace output.
- The trace includes a mix of addresses and data values, indicating the format and structure used in dump record traces.

**Example:**

- **Address:** 0000 08500000
- **Data:** C4F8F1F1

**Explanation:**

- The address 0000 08500000 refers to a specific memory location.
- The data value C4F8F1F1 is stored at that address.

The trace continues in a similar fashion, displaying various addresses and data values. The trace concludes with a statement indicating the completion of the dump record trace.
The Dump Record Trace

Field description

The left-hand side of the trace shows a full hexadecimal dump of the record and the section on the right shows the same data in character format. The following table describes selected fields in “Dump Record Trace Example” on page 39-1.

Table 39-1. Field Description

<table>
<thead>
<tr>
<th>Label</th>
<th>Hexadecimal</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0850</td>
<td>Record Length</td>
</tr>
<tr>
<td>B</td>
<td>0000</td>
<td>Binary zeros</td>
</tr>
<tr>
<td>C</td>
<td>C4D7D4</td>
<td>Literal identifier ‘DPM’</td>
</tr>
<tr>
<td>D</td>
<td>83</td>
<td>DB2 release number</td>
</tr>
<tr>
<td>E</td>
<td>00000850</td>
<td>Full record length</td>
</tr>
<tr>
<td>F</td>
<td>D7D4D6F1 C4F8F1F1 40404040 40404040</td>
<td>Location</td>
</tr>
<tr>
<td>G</td>
<td>40404040 40404040</td>
<td>Data sharing group name</td>
</tr>
<tr>
<td>H</td>
<td>C4F8F1F1</td>
<td>Subsystem identifier</td>
</tr>
<tr>
<td>I</td>
<td>40404040 40404040</td>
<td>Member name</td>
</tr>
<tr>
<td>J</td>
<td>BC2B07BB B05BF1A0</td>
<td>Sort timestamp</td>
</tr>
<tr>
<td>K</td>
<td>BC2B07BB B05BF1A0</td>
<td>TIMEZONE adjusted timestamp</td>
</tr>
<tr>
<td>L</td>
<td>D7C1E540 40404040 40404040</td>
<td>Correlation name</td>
</tr>
<tr>
<td>M</td>
<td>40404040 40404040</td>
<td>Correlation number</td>
</tr>
<tr>
<td>N</td>
<td>E3E2D640 40404040</td>
<td>Connecting system type</td>
</tr>
<tr>
<td>O</td>
<td>00100000</td>
<td>Flags: record type, correlation data present, CPU data present and DDF data present</td>
</tr>
<tr>
<td>P</td>
<td>0003</td>
<td>IFCID</td>
</tr>
<tr>
<td>Q</td>
<td>C45C9C2 D4C9D7E2 C9D7E2C1 D8F8F1F1 BC2AFF59 81F30001</td>
<td>Logical unit of work ID (LUWID): network ID, LU name, instance number and commit count</td>
</tr>
<tr>
<td>R</td>
<td>00000270</td>
<td>Pointer to the first self-defining section</td>
</tr>
</tbody>
</table>

“Column Descriptions of the Dump Record Trace” on page 39-3
The column description of the Dump record trace.

“ACE Cross-Reference Table” on page 39-5
For every trace specified, an ACE cross-reference table is printed at the end of each location.

“Field Descriptions of the Dump Record Trace” on page 39-6
This topic describes the general format of the IFCID records presented in the dump record trace.
### Column Descriptions of the Dump Record Trace

The following columns are shown on the dump record trace:

**PRIMAUTH**
The authorization ID under which the transaction is running. Derived from the DB2 field QWHCAID.

**ORIGAUTH**
The original authorization ID under which the transaction started. Derived from the DB2 field QWHCOPID.

**PLANNAME**
The DB2 plan name. Derived from the DB2 field QWHCPLAN.

**CONNECT**
The connection ID. Derived from the DB2 field QWHCCN.

**CORRNAME**
The correlation name. Derived from the DB2 field QWHCCV.

**CORRNMBR**
The correlation number. Derived from the DB2 field QWHCCV.

**CONNTYPE**
The type of connection being used to interface with DB2. Derived from the DB2 field QWHCCST.

**INSTANCE**
The unique number assigned to a thread. Derived from the DB2 field QWHSUNIQ.

**RECORD TIME**
The timestamp contained in the trace record. The format is hours, minutes, seconds, and hundred-millionths of a second. Derived from the DB2 field QWHSSTCK.

**TCB CPU TIME**
The CPU time stored in the trace record. The format is minutes, seconds, and hundred-millionths of a second. Derived from the DB2 field QWHUCPU.

**DEST SEQ NO**
The destination sequence number. Derived from the DB2 field QWHSWSEQ.

**ACE**
The relative number of the agent control element address in the ACE cross-reference table. This table is printed at the end of each location for every trace specified. Derived from the DB2 field QWHSACE.

**IFCID**
The instrumentation facility component identification (DB2 trace record type). Derived from the DB2 field QWHSIID.

**DESCRIPTION**
A brief description of the IFCID record. The description indicates whether the record contains accounting, statistics, or performance data. For performance data, the description also indicates the event.

**DATA**
The data is printed in the standard hexadecimal dump format. The character format is on the right.
Column Descriptions

Note: The self-defining section starts at offset x'01E8'. The IFC data sections start at offset x'0248'.
For every trace specified, an ACE cross-reference table is printed at the end of each location.

The columns of the ACE cross-reference table are:

**ACE NUMBER**
The cross-reference number for the hexadecimal address of the agent control element. The lowest valid cross-reference number is 1. 0 indicates that the ACE address is not available.

**ACE ADDRESS**
The hexadecimal address of the agent control element. Derived from the DB2 field QWHSACE.
Field Descriptions

Field Descriptions of the Dump Record Trace

This topic describes the general format of the IFCID records presented in the dump record trace.

The records are presented in the requested sequence. There is one entry on the report for each record selected from the input data set, therefore, the report can show more than one record of the same IFCID record type. Use the record time field on the report to distinguish between records with the same IFCID record type.

Note:
1. An arrow (→) pointing to the right on the trace indicates the start of an event.
2. An arrow (←) pointing to the left on the trace indicates the end of an event.

"Logical Unit of Work Identifiers" on page 39-7
The logical unit of work identifiers are shown in the DATA column in front of the formatted data.

"DDF Data" on page 39-8
DDF data is only printed if there is a DDF header.

"Accounting Token" on page 39-9
All record trace reports show the value (in hexadecimal) of the accounting token in the DATA column when it contains a value other than blanks or binary zeros.
Logical Unit of Work Identifiers

The logical unit of work identifiers are shown in the DATA column in front of the formatted data.

**NETWORKID:** USIBMSY  **LUNAME:** SY30BDB2  **LUWSEQ:** 2

**Field description**

Here is a description of the field labels shown in the previous sample:

**NETWORKID**

The network identifier. Derived from the DB2 field QWHSNID.

**LUNAME**

The logical unit name. Derived from the DB2 field QWHSLUNM.

**LUWSEQ**

The sequence number of the logical unit of work. Derived from the DB2 field QWHSLUCC.
Logical Unit of Work Identifiers

**DDF Data**

DDF data is only printed if there is a DDF header.

The following sample is printed in the DATA column after the record:

REQUESTING LOCATION: USIBMSYSTDB2
REQUESTING TIMESTAMP: 01/30/10 18:54:53.90530718
AR NAME: USIBMSYSTDB2     PRDID: DB2 V9 R1 M0

**Field description**

Here is a description of the field labels shown in the previous sample:

**REQUESTING LOCATION**
The location requesting the work. Derived from the DB2 field QWHDRQNM.

**REQUESTING TIMESTAMP**
The timestamp of the requester location. Derived from the DB2 field QWHDTSTP.

**AR NAME**
The name of the application requester. Derived from the DB2 field QWHDSVNM. QWHDSVNM is the SRVNAM parameter of the DDM exchange server attributes command (EXCSAT).

**PRDID**
The name, version, release, and modification level of the product making the request. Derived from the DB2 field QWHDPRID.
Logical Unit of Work Identifiers

Accounting Token

All record trace reports show the value (in hexadecimal) of the accounting token in the DATA column when it contains a value other than blanks or binary zeros.

The accounting token is used to correlate CICS records with DB2 records for the same task. If TOKENI=YES for TYPE=INIT, TOKENE=YES for TYPE=ENTRY, or both applies, in the resource control table, then the CICS logical unit of work ID (LUWID) minus the commit count (2 bytes) is passed to this field.

ACCTKN X'00000000000000000000000000000000000000000000'

The first 8 bytes contain the network name, and the following 8 bytes contain the LU name. The final 6 bytes are the unique value.
Chapter 40. IFCID Record Blocks

This topic describes the Instrumentation Facility Component Identifier (IFCID) record trace blocks. The description within each block is presented in alphabetical order.

“IFCID 001 - System Statistics” on page 40-19
System service statistics are written at regular intervals specified by the install parameter STATISTICS TIME on panel DSNTIPN.

“IFCID 002 - DB2 Statistics” on page 40-63

“IFCID 003 - Accounting” on page 40-144

“IFCID 004 - Trace Start” on page 40-208
This topic shows detailed information about “Record Trace - IFCID 004 - Trace Start”.

“IFCID 005 - Trace Stop” on page 40-209
This topic shows detailed information about “Record Trace - IFCID 005 - Trace Stop”.

“IFCID 006 - Read I/O Start” on page 40-210
This topic shows detailed information about “Record Trace - IFCID 006 - Read I/O Start”.

“IFCID 007 - Read I/O Stop” on page 40-212
This topic shows detailed information about “Record Trace - IFCID 007 - Read I/O Stop”.

“IFCID 008 - Write I/O Synch” on page 40-213
This topic shows detailed information about “Record Trace - IFCID 008 - Write I/O Synch”.

“IFCID 009 - Write I/O” on page 40-215
This topic shows detailed information about “Record Trace - IFCID 009 - Write I/O”.

“IFCID 010 - Write I/O Asynch” on page 40-216
This topic shows detailed information about “Record Trace - IFCID 010 - Write I/O Asynch”.

“IFCID 011 - Validate Exit” on page 40-218
This topic shows detailed information about “Record Trace - IFCID 011 - Validate Exit”.

“IFCID 012 - Edit Exit to Encode” on page 40-219
This topic shows detailed information about “Record Trace - IFCID 012 - Edit Exit to Encode”.

“IFCID 013 - Hash Scan Input Start” on page 40-220
This topic shows detailed information about “Record Trace - IFCID 013 - Hash Scan Input Start”.

“IFCID 014 - Hash Scan End” on page 40-222
This topic shows detailed information about “Record Trace - IFCID 014 - Hash Scan End”.

“IFCID 015 - Index Scan Begin” on page 40-223
This topic shows detailed information about “Record Trace - IFCID 015 - Index Scan Begin”.

“IFCID 016 - Insert Scan Begin” on page 40-225
This topic shows detailed information about “Record Trace - IFCID 016 - Insert Scan Begin”.

© Copyright IBM Corp. 1985, 2013
“IFCID 017 - Sequential Scan Begin” on page 40-227
This topic shows detailed information about “Record Trace - IFCID 017 - Sequential Scan Begin”.

“IFCID 018 - Scan End” on page 40-229
This topic shows detailed information about “Record Trace - IFCID 018 - Scan End”.

“IFCID 019 - Edit Exit to Decode” on page 40-231
This topic shows detailed information about “Record Trace - IFCID 019 - Edit Exit to Decode”.

“IFCID 020 - Lock Summary” on page 40-232
This topic shows detailed information about “Record Trace - IFCID 020 - Lock Summary”.

“IFCID 021 - Lock Detail” on page 40-234
This topic shows detailed information about “Record Trace - IFCID 021 - Lock Detail”.

“IFCID 022 - Minibind” on page 40-239
This topic shows detailed information about “Record Trace - IFCID 022 - Minibind”.

“IFCID 023 - Utility Start” on page 40-248
This topic shows detailed information about “Record Trace - IFCID 023 - Utility Start”.

“IFCID 024 - Utility Change” on page 40-252
This topic shows detailed information about “Record Trace - IFCID 024 - Utility Change”.

“IFCID 025 - Utility End” on page 40-254
This topic shows detailed information about “Record Trace - IFCID 025 - Utility End”.

“IFCID 026 - IBM Service Record” on page 40-257
This topic shows detailed information about “Record Trace - IFCID 026 - IBM Service Record”.

“IFCID 027 - Sort Workfile Records” on page 40-258
This topic shows detailed information about “Record Trace - IFCID 027 - Sort Workfile Records”.

“IFCID 028 - Sort Phase Detail” on page 40-260
This topic shows detailed information about “Record Trace - IFCID 028 - Sort Phase Detail”.

“IFCID 029 - EDM Request Start” on page 40-262
This topic shows detailed information about “Record Trace - IFCID 029 - EDM Request Start”.

“IFCID 030 - EDM Request End” on page 40-264
This topic shows detailed information about “Record Trace - IFCID 030 - EDM Request End”.

“IFCID 031 - EDM Full” on page 40-266
This topic shows detailed information about “Record Trace - IFCID 031 - EDM Full”.

“IFCID 032 - Log Wait Start” on page 40-269
This topic shows detailed information about “Record Trace - IFCID 032 - Log Wait Start”.

“IFCID 033 - IBM Service Record” on page 40-270
This topic shows detailed information about “Record Trace - IFCID 033 - IBM Service Record”.

40-2 OMEGAMON XE for DB2 PE & PM: Report Reference
“IFCID 034 - Log Read Start” on page 40-271
This topic shows detailed information about “Record Trace - IFCID 034 - Log Read Start”.

“IFCID 035 - Log Read End” on page 40-272
This topic shows detailed information about “Record Trace - IFCID 035 - Log Read End”.

“IFCID 036 - Log Non I/O Start” on page 40-273
This topic shows detailed information about “Record Trace - IFCID 036 - Log Non I/O Start”.

“IFCID 037 - Log Non I/O End” on page 40-274
This topic shows detailed information about “Record Trace - IFCID 037 - Log Non I/O End”.

“IFCID 038 - Active Write Start” on page 40-275
This topic shows detailed information about “Record Trace - IFCID 038 - Active Write Start”.

“IFCID 039 - Active Write End” on page 40-276
This topic shows detailed information about “Record Trace - IFCID 039 - Active Write End”.

“IFCID 040 - Archive Write Start” on page 40-277
This topic shows detailed information about “Record Trace - IFCID 040 - Archive Write Start”.

“IFCID 041 - Archive Write End” on page 40-278
This topic shows detailed information about “Record Trace - IFCID 041 - Archive Write End”.

“IFCID 042 - Checkpoint Start” on page 40-279
This topic shows detailed information about “Record Trace - IFCID 042 - Checkpoint Start”.

“IFCID 043 - Checkpoint End” on page 40-280
This topic shows detailed information about “Record Trace - IFCID 043 - Checkpoint End”.

“IFCID 044 - Lock Suspend” on page 40-281
This topic shows detailed information about “Record Trace - IFCID 044 - Lock Suspend”.

“IFCID 045 - Lock Resume” on page 40-285
This topic shows detailed information about “Record Trace - IFCID 045 - Lock Resume”.

“IFCID 046 - IBM Service Record” on page 40-287
This topic shows detailed information about “Record Trace - IFCID 046 - IBM Service Record”.

“IFCID 047 - IBM Service Record” on page 40-288
This topic shows detailed information about “Record Trace - IFCID 047 - IBM Service Record”.

“IFCID 048 - IBM Service Record” on page 40-289
This topic shows detailed information about “Record Trace - IFCID 048 - IBM Service Record”.

“IFCID 049 - IBM Service Record” on page 40-290
This topic shows detailed information about “Record Trace - IFCID 049 - IBM Service Record”.

“IFCID 050 - IBM Service Record” on page 40-291
This topic shows detailed information about “Record Trace - IFCID 050 - IBM Service Record”. 
This topic shows detailed information about “Record Trace - IFCID 069 - IBM Service Record”.

This topic shows detailed information about “Record Trace - IFCID 070 - Commit Phase 2 Start”.

This topic shows detailed information about “Record Trace - IFCID 071 - IBM Service Record”.

This topic shows detailed information about “Record Trace - IFCID 072 - Create Thread Start”.

This topic shows detailed information about “Record Trace - IFCID 073 - Create Thread End”.

This topic shows detailed information about “Record Trace - IFCID 074 - Terminate Thread Start”.

This topic shows detailed information about “Record Trace - IFCID 075 - Terminate Thread End”.

This topic shows detailed information about “Record Trace - IFCID 076 - End of Memory Start”.

This topic shows detailed information about “Record Trace - IFCID 077 - End of Memory End”.

This topic shows detailed information about “Record Trace - IFCID 078 - End of Task Start”.

This topic shows detailed information about “Record Trace - IFCID 079 - End of Task End”.

This topic shows detailed information about “Record Trace - IFCID 080 - IBM Service Record”.

This topic shows detailed information about “Record Trace - IFCID 081 - IBM Service Record”.

This topic shows detailed information about “Record Trace - IFCID 082 - Identify Start”.

This topic shows detailed information about “Record Trace - IFCID 083 - Identify End”.

This topic shows detailed information about “Record Trace - IFCID 084 - Prepare Start”.

This topic shows detailed information about “Record Trace - IFCID 085 - Prepare End”.  

Chapter 40. IFCID Record Blocks  40-5
“IFCID 086 - Signon Start” on page 40-338
This topic shows detailed information about “Record Trace - IFCID 086 - Signon Start”.

“IFCID 087 - Signon End” on page 40-339
This topic shows detailed information about “Record Trace - IFCID 087 - Signon End”.

“IFCID 088 - Synch Start” on page 40-340
This topic shows detailed information about “Record Trace - IFCID 088 - Synch Start”.

“IFCID 089 - Synch End” on page 40-341
This topic shows detailed information about “Record Trace - IFCID 089 - Synch End”.

“IFCID 090 - DB2 Command Start” on page 40-342
This topic shows detailed information about “Record Trace - IFCID 090 - DB2 Command Start”.

“IFCID 091 - Command End” on page 40-343
This topic shows detailed information about “Record Trace - IFCID 091 - Command End”.

“IFCID 092 - AMS Command Start” on page 40-344
This topic shows detailed information about “Record Trace - IFCID 092 - AMS Command Start”.

“IFCID 093 - IBM Service Record” on page 40-345
This topic shows detailed information about “Record Trace - IFCID 093 - IBM Service Record”.

“IFCID 094 - IBM Service Record” on page 40-346
This topic shows detailed information about “Record Trace - IFCID 094 - IBM Service Record”.

“IFCID 095 - Sort Start” on page 40-347
This topic shows detailed information about “Record Trace - IFCID 095 - Sort Start”.

“IFCID 096 - Sort End” on page 40-348
This topic shows detailed information about “Record Trace - IFCID 096 - Sort End”.

“IFCID 097 - AMS Command End” on page 40-351
This topic shows detailed information about “Record Trace - IFCID 097 - AMS Command End”.

“IFCID 098 - IBM Service Record” on page 40-352
This topic shows detailed information about “Record Trace - IFCID 098 - IBM Service Record”.

“IFCID 099 - IBM Service Record” on page 40-353
This topic shows detailed information about “Record Trace - IFCID 099 - IBM Service Record”.

“IFCID 100 - IBM Service Record” on page 40-354
This topic shows detailed information about “Record Trace - IFCID 100 - IBM Service Record”.

“IFCID 101 - IBM Service Record” on page 40-355
This topic shows detailed information about “Record Trace - IFCID 101 - IBM Service Record”.

“IFCID 102 - IBM Service Record” on page 40-356
This topic shows detailed information about “Record Trace - IFCID 102 - IBM Service Record”.
<table>
<thead>
<tr>
<th>IFCID 151 - User Record</th>
<th>on page 40-528</th>
</tr>
</thead>
<tbody>
<tr>
<td>This topic shows detailed information about “Record Trace - IFCID 151 - User Record”.</td>
<td></td>
</tr>
<tr>
<td>IFCID 152 - User Record</td>
<td>on page 40-529</td>
</tr>
<tr>
<td>This topic shows detailed information about “Record Trace - IFCID 152 - User Record”.</td>
<td></td>
</tr>
<tr>
<td>IFCID 153 - User Record</td>
<td>on page 40-530</td>
</tr>
<tr>
<td>This topic shows detailed information about “Record Trace - IFCID 153 - User Record”.</td>
<td></td>
</tr>
<tr>
<td>IFCID 154 - User Record</td>
<td>on page 40-531</td>
</tr>
<tr>
<td>This topic shows detailed information about “Record Trace - IFCID 154 - User Record”.</td>
<td></td>
</tr>
<tr>
<td>IFCID 155 - User Record</td>
<td>on page 40-532</td>
</tr>
<tr>
<td>This topic shows detailed information about “Record Trace - IFCID 155 - User Record”.</td>
<td></td>
</tr>
<tr>
<td>IFCID 156 - User Record</td>
<td>on page 40-533</td>
</tr>
<tr>
<td>This topic shows detailed information about “Record Trace - IFCID 156 - User Record”.</td>
<td></td>
</tr>
<tr>
<td>IFCID 157 - DRDS RDS Interface</td>
<td>on page 40-534</td>
</tr>
<tr>
<td>This topic shows detailed information about “Record Trace - IFCID 157 - DRDS RDS Interface”.</td>
<td></td>
</tr>
<tr>
<td>IFCID 158 - DRDS CNV Interface</td>
<td>on page 40-535</td>
</tr>
<tr>
<td>This topic shows detailed information about “Record Trace - IFCID 158 - DRDS CNV Interface”.</td>
<td></td>
</tr>
<tr>
<td>IFCID 159 - DRDS Req Site Data</td>
<td>on page 40-536</td>
</tr>
<tr>
<td>This topic shows detailed information about “Record Trace - IFCID 159 - DRDS Req Site Data”.</td>
<td></td>
</tr>
<tr>
<td>IFCID 160 - DC Requester</td>
<td>on page 40-537</td>
</tr>
<tr>
<td>This topic shows detailed information about “Record Trace - IFCID 160 - DC Requester”.</td>
<td></td>
</tr>
<tr>
<td>IFCID 161 - DC Server</td>
<td>on page 40-538</td>
</tr>
<tr>
<td>This topic shows detailed information about “Record Trace - IFCID 161 - DC Server”.</td>
<td></td>
</tr>
<tr>
<td>IFCID 162 - DTM Request</td>
<td>on page 40-539</td>
</tr>
<tr>
<td>This topic shows detailed information about “Record Trace - IFCID 162 - DTM Request”.</td>
<td></td>
</tr>
<tr>
<td>IFCID 163 - DTM Respond</td>
<td>on page 40-540</td>
</tr>
<tr>
<td>This topic shows detailed information about “Record Trace - IFCID 163 - DTM Respond”.</td>
<td></td>
</tr>
<tr>
<td>IFCID 164 - IBM Service Record</td>
<td>on page 40-541</td>
</tr>
<tr>
<td>This topic shows detailed information about “Record Trace - IFCID 164 - IBM Service Record”.</td>
<td></td>
</tr>
<tr>
<td>IFCID 165 - IBM Service Record</td>
<td>on page 40-542</td>
</tr>
<tr>
<td>This topic shows detailed information about “Record Trace - IFCID 165 - IBM Service Record”.</td>
<td></td>
</tr>
<tr>
<td>IFCID 166 - IBM Service Record</td>
<td>on page 40-543</td>
</tr>
<tr>
<td>This topic shows detailed information about “Record Trace - IFCID 166 - IBM Service Record”.</td>
<td></td>
</tr>
<tr>
<td>IFCID 167 - Conv Alloc Req Queued</td>
<td>on page 40-544</td>
</tr>
<tr>
<td>This topic shows detailed information about “Record Trace - IFCID 167 - Conv Alloc Req Queued”.</td>
<td></td>
</tr>
</tbody>
</table>
Chapter 40. IFCID Record Blocks

“IFCID 188 - READs Data Capture End” on page 40-578
This topic shows detailed information about “Record Trace - IFCID 188 - READs Data Capture End”.

“IFCID 190 - IBM Service Record” on page 40-580
This topic shows detailed information about “Record Trace - IFCID 190 - IBM Service Record”.

“IFCID 191 - DDM Level 6B Objects” on page 40-581
“IFCID 191 - 6B DSS Section” on page 40-588
This topic shows detailed information about “Record Trace - IFCID 191 - 6B DSS Section”.

“IFCID 192 - DDM Level 6A Header Errors” on page 40-589
“IFCID 193 - UOW/SQLCODE Mismatch” on page 40-593
This topic shows detailed information about “Record Trace - IFCID 193 - UOW/SQLCODE Mismatch”.

“IFCID 194 - Invalid SNA FMH-5 Received” on page 40-594
This topic shows detailed information about “Record Trace - IFCID 194 - Invalid SNA FMH-5 Received”.

“IFCID 195 - SQLDA Discrepancy” on page 40-595
This topic shows detailed information about “Record Trace - IFCID 195 - SQLDA Discrepancy”.

“IFCID 196 - Timeout Data” on page 40-597
“IFCID 197 - DB2 Messages” on page 40-603
This topic shows detailed information about “Record Trace - IFCID 197 - DB2 Messages”.

“IFCID 198 - Buffer Manager Page Access” on page 40-604
This topic shows detailed information about “Record Trace - IFCID 198 - Buffer Manager Page Access”.

“IFCID 199 - Buffer Pool Statistics at Data Set Level” on page 40-606
This topic shows detailed information about “Record Trace - IFCID 199 - Buffer Pool Statistics at Data Set Level”.

“IFCID 201 - Alter Buffer Pool” on page 40-609
This topic shows detailed information about “Record Trace - IFCID 201 - Alter Buffer Pool”.

“IFCID 202 - Buffer Pool Attributes” on page 40-612
This topic shows detailed information about “Record Trace - IFCID 202 - Buffer Pool Attributes”.

“IFCID 203 - DDF Heuristic COMMIT/ROLLBK” on page 40-615
This topic shows detailed information about “Record Trace - IFCID 203 - DDF Heuristic COMMIT/ROLLBK”.

“IFCID 204 - DDF Partner Cold Start” on page 40-617
This topic shows detailed information about “Record Trace - IFCID 204 - DDF Partner Cold Start”.

“IFCID 205 - DDF Warm Start Log Name Error Information” on page 40-619
“IFCID 206 - DDF Protocol Error” on page 40-623
This topic shows detailed information about “Record Trace - IFCID 206 - DDF Protocol Error”.

“IFCID 207 - DDF Heuristic Damage” on page 40-625
This topic shows detailed information about “Record Trace - IFCID 207 - DDF Heuristic Damage”.
“IFCID 208 - DDF Syncpoint Protocol Error” on page 40-627
This topic shows detailed information about “Record Trace - IFCID 208 - DDF Syncpoint Protocol Error”.

“IFCID 209 - DDF Syncpoint Comm Failure” on page 40-629
This topic shows detailed information about “Record Trace - IFCID 209 - DDF Syncpoint Comm Failure”.

“IFCID 210 - Warm Start Log Name Change” on page 40-631
This topic shows detailed information about “Record Trace - IFCID 210 - Warm Start Log Name Change”.

“IFCID 211 - Claim Data” on page 40-632
This topic shows detailed information about “Record Trace - IFCID 211 - Claim Data”.

“IFCID 212 - Drain Data” on page 40-633
This topic shows detailed information about “Record Trace - IFCID 212 - Drain Data”.

“IFCID 213 - Drain Lock Wait Start” on page 40-635
This topic shows detailed information about “Record Trace - IFCID 213 - Drain Lock Wait Start”.

“IFCID 214 - Drain Lock Wait End” on page 40-637
This topic shows detailed information about “Record Trace - IFCID 214 - Drain Lock Wait End”.

“IFCID 215 - Claim Count 0 Wait Start” on page 40-639
This topic shows detailed information about “Record Trace - IFCID 215 - Claim Count 0 Wait Start”.

“IFCID 216 - Claim Count 0 Wait End” on page 40-640
This topic shows detailed information about “Record Trace - IFCID 216 - Claim Count 0 Wait End”.

“IFCID 217 - Storage Pools” on page 40-641

“IFCID 218 - Lock Avoidance Summary” on page 40-651
This topic shows detailed information about “Record Trace - IFCID 218 - Lock Avoidance Summary”.

“IFCID 219 - Utility LISTDEF List Information” on page 40-652
This topic shows detailed information about “Record Trace - IFCID 219 - Utility LISTDEF List Information”.

“IFCID 220 - Utility Data Set Information” on page 40-653
This topic shows detailed information about “Record Trace - IFCID 220 - Utility Data Set Information”.

“IFCID 221 - Parallel Group Execution” on page 40-655

“IFCID 222 - Parallel Group Elapsed Time” on page 40-662
This topic shows detailed information about “Record Trace - IFCID 222 - Parallel Group Elapsed Time”.

“IFCID 223 - Lock Avoidance Detail” on page 40-664
This topic shows detailed information about “Record Trace - IFCID 223 - Lock Avoidance Detail”.

“IFCID 224 - Select Procedure Bypassed” on page 40-665
This topic shows detailed information about “Record Trace - IFCID 224 - Select Procedure Bypassed”.

“IFCID 225 - System Storage Usage” on page 40-666

“IFCID 226 - Page Latch Contention Start” on page 40-682
This topic shows detailed information about “Record Trace - IFCID 226 - Page Latch Contention Start”.
This topic shows detailed information about “Record Trace - IFCID 227 - Page Latch Contention End”.

This topic shows detailed information about “Record Trace - IFCID 228 - Archive Deallocation Start”.

This topic shows detailed information about “Record Trace - IFCID 229 - Archive Deallocation End”.

This topic shows detailed information about “Record Trace - IFCID 230 - Group Buffer Pool Attributes”.

This topic shows detailed information about “Record Trace - IFCID 231 - Parallel Group Task Time”.

This topic shows detailed information about “Record Trace - IFCID 233 - Call User Routine”.

This topic shows detailed information about “Record Trace - IFCID 234 - Calling Agent Auth IDs”.

This topic shows detailed information about “Record Trace - IFCID 236 - DDF SNA XLN Protocol Error”.

This topic shows detailed information about “Record Trace - IFCID 237 - Set Current Degree”.

This topic shows detailed information about “Record Trace - IFCID 238 - IBM Service Record”.

This topic shows detailed information about “Record Trace - IFCID 239 - Overflow Package/DBRM”.

This topic shows detailed information about “Record Trace - IFCID 247 - SQLDA Data and Input Host Variable Data”.

This topic shows detailed information about “Record Trace - IFCID 248 - IBM Service Record”.

This topic shows detailed information about “Record Trace - IFCID 249 - EDM Pool Invalidate DBD”.

This topic shows detailed information about “Record Trace - IFCID 250 - Connect/Rebuild Connect/Disconnect Group Bpool”.

This topic shows detailed information about “Record Trace - IFCID 251 - Buffer Manager PSET/Part P-Lock Request”.

This topic shows detailed information about “Record Trace - IFCID 252 - IBM Service Record”.
“IFCID 254 - Coupling Facility Cache Structure Statistics” on page 40-733
This topic shows detailed information about “Record Trace - IFCID 254 - Coupling Facility Cache Structure Statistics”.

“IFCID 255 - Buffer Refresh Due to XI” on page 40-736
This topic shows detailed information about “Record Trace - IFCID 255 - Buffer Refresh Due to XI”.

“IFCID 256 - Alter Group Buffer Pool” on page 40-738
This topic shows detailed information about “Record Trace - IFCID 256 - Alter Group Buffer Pool”.

“IFCID 257 - IRLM Notify Req Detail” on page 40-740
This topic shows detailed information about “Record Trace - IFCID 257 - IRLM Notify Req Detail”.

“IFCID 258 - Data Set Extend Activity” on page 40-743
This topic shows detailed information about “Record Trace - IFCID 258 - Data Set Extend Activity”.

“IFCID 259 - Buffer Manager Pg P-Lock Req” on page 40-745
This topic shows detailed information about “Record Trace - IFCID 259 - Buffer Manager Pg P-Lock Req”.

“IFCID 260 - IBM Service Record” on page 40-747
This topic shows detailed information about “Record Trace - IFCID 260 - IBM Service Record”.

“IFCID 261 - Group Buffer Pool Checkpoint” on page 40-748
This topic shows detailed information about “Record Trace - IFCID 261 - Group Buffer Pool Checkpoint”.

“IFCID 262 - GBPOOLT Castout Threshold Processing” on page 40-750
This topic shows detailed information about “Record Trace - IFCID 262 - GBPOOLT Castout Threshold Processing”.

“IFCID 263 - Page Set and Partition Castout Detail” on page 40-752
This topic shows detailed information about “Record Trace - IFCID 263 - Page Set and Partition Castout Detail”.

“IFCID 265 - IBM Service Record” on page 40-755
This topic shows detailed information about “Record Trace - IFCID 265 - IBM Service Record”.

“IFCID 266 - IBM Service Record” on page 40-756
This topic shows detailed information about “Record Trace - IFCID 266 - IBM Service Record”.

“IFCID 267 - CF Rebuild/Alter/Start” on page 40-757
This topic shows detailed information about “Record Trace - IFCID 267 - CF Rebuild/Alter/Start”.

“IFCID 268 - CF Rebuild/Alter End” on page 40-758
This topic shows detailed information about “Record Trace - IFCID 268 - CF Rebuild/Alter End”.

“IFCID 269 - Trusted/Context Trace” on page 40-761
This topic shows detailed information about “Record Trace - IFCID 269 - Trusted/Context Trace”.

“IFCID 270 - Trusted/Context Trace” on page 40-764
This topic shows detailed information about “Record Trace - IFCID 270 - Trusted/Context Trace”.

“IFCID 271 - Row Level and Column Level Access Control” on page 40-765
This topic shows detailed information about “Record Trace - IFCID 271 - Row Level and Column Level Access Control”.
"IFCID 272 - Associate Locators" on page 40-767
This topic shows detailed information about "Record Trace - IFCID 272 - Associate Locators".

"IFCID 273 - Allocate Cursor" on page 40-769
This topic shows detailed information about "Record Trace - IFCID 273 - Allocate Cursor".

"IFCID 305 - Table Check Constraint" on page 40-771
This topic shows detailed information about "Record Trace - IFCID 305 - Table Check Constraint".

"IFCID 311 - Global Temp Table Usage" on page 40-773
This topic shows detailed information about "Record Trace - IFCID 311 - Global Temp Table Usage".

"IFCID 313 - Uncommitted Unit of Recovery" on page 40-775
This topic shows detailed information about "Record Trace - IFCID 313 - Uncommitted Unit of Recovery".

"IFCID 314 - Authorization Exit Parameters" on page 40-777
This topic shows detailed information about "Record Trace - IFCID 314 - Authorization Exit Parameters".

"IFCID 316 - SQL Statement Statistics" on page 40-779
This topic shows detailed information about "Record Trace - IFCID 316 - SQL Statement Statistics".

"IFCID 317 - SQL Statement String" on page 40-787
This topic shows detailed information about "Record Trace - IFCID 317 - SQL Statement String".

"IFCID 319 - Audit Security Record" on page 40-788
This topic shows detailed information about "Record Trace - IFCID 319 - Audit Security Record".

"IFCID 321 - Force-at-Commit Begin" on page 40-790
This topic shows detailed information about "Record Trace - IFCID 321 - Force-at-Commit Begin".

"IFCID 322 - Force-at-Commit End" on page 40-791
This topic shows detailed information about "Record Trace - IFCID 322 - Force-at-Commit End".

"IFCID 324 - Function Resolution" on page 40-792
This topic shows detailed information about "Record Trace - IFCID 324 - Function Resolution".

"IFCID 325 - Trigger Activation" on page 40-794
This topic shows detailed information about "Record Trace - IFCID 325 - Trigger Activation".

"IFCID 329 - IXL Suspensions" on page 40-796
This topic shows detailed information about "Record Trace - IFCID 329 - IXL Suspensions".

"IFCID 330 - Active Log Space Shortage" on page 40-797
This topic shows detailed information about "Record Trace - IFCID 330 - Active Log Space Shortage".

"IFCID 331 - IBM Service Record" on page 40-798
This topic shows detailed information about "Record Trace - IFCID 331 - IBM Service Record".

"IFCID 332 - IBM Service Record" on page 40-799
This topic shows detailed information about "Record Trace - IFCID 332 - IBM Service Record".
"IFCID 333 - IBM Service Record" on page 40-800
This topic shows detailed information about "Record Trace - IFCID 333 - IBM Service Record".

"IFCID 335 - System Event Stalled" on page 40-801
This topic shows detailed information about "Record Trace - IFCID 335 - System Event Stalled".

"IFCID 342 - WF/TEMP DB Usage" on page 40-802
This topic shows detailed information about "Record Trace - IFCID 342 - WF/TEMP DB Usage".

"IFCID 343 - MAXTEMPS Limit/Exceeded" on page 40-804
This topic shows detailed information about "Record Trace - IFCID 343 - MAXTEMPS Limit/Exceeded".

"IFCID 345 - Trace Data / SP/UDF" on page 40-805
This topic shows detailed information about "Record Trace - IFCID 345 - Trace Data / SP/UDF".

"IFCID 346 - Package/DBRM Detail" on page 40-806
This topic shows detailed information about "Record Trace - IFCID 346 - Package/DBRM Detail".

"IFCID 350 - SQL Statement" on page 40-807
This topic shows detailed information about "Record Trace - IFCID 350 - SQL Statement".

"IFCID 351 - Wait TCPIP LOB" on page 40-809
This topic shows detailed information about "Record Trace - IFCID 351 - Wait TCPIP LOB".

"IFCID 353 - IBM Service Record" on page 40-810
This topic shows detailed information about "Record Trace - IFCID 353 - IBM Service Record".

"IFCID 354 - IBM Service Record" on page 40-811
This topic shows detailed information about "Record Trace - IFCID 354 - IBM Service Record".

"IFCID 357 - Beginning of an Index I/O Parallel INSERT" on page 40-812
This topic shows detailed information about "Record Trace - IFCID 357 - Beginning of an Index I/O Parallel INSERT".

"IFCID 358 - End of an Index I/O Parallel INSERT" on page 40-813
This topic shows detailed information about "Record Trace - IFCID 358 - End of an Index I/O Parallel INSERT".

"IFCID 359 - Index Page Split" on page 40-814
This topic shows detailed information about "Record Trace - IFCID 359 - Index Page Split".

"IFCID 361 - Audit Admin Authorities" on page 40-815
This topic shows detailed information about "Record Trace - IFCID 361 - Audit Admin Authorities".

"IFCID 362 - Start Trace and Stop Trace with Audit Policy" on page 40-818
This topic shows detailed information about "Record Trace - IFCID 362 - Start Trace and Stop Trace with Audit Policy".

"IFCID 363 - Parallel Straw Model Performance Trace" on page 40-820

"IFCID 365 - Remote Location Statistics" on page 40-825
This topic shows detailed information about "Record Trace - IFCID 365 - Remote Location Statistics".

"IFCID 366 - Incompatible Function Char(DEC) Executed" on page 40-828
This topic shows detailed information about "Record Trace - IFCID 366 - Incompatible Function Char(DEC) Executed".
System service statistics are written at regular intervals specified by the install parameter STATISTICS TIME on panel DSNTIPN.

Most counters in this record are accumulated since DB2 was last started. Some counters can include values recorded prior to the report period covered. Values are reset to zero when DB2 is started.

This topic shows detailed information about “Record Trace - IFCID 001 - Checkpoint and IFI Data”.

This topic shows detailed information about “Record Trace - IFCID 001 - CPU Time Data”.

This topic shows detailed information about “Record Trace - IFCID 001 - DB2 Command Data”.

This topic shows detailed information about “Record Trace - IFCID 001 - DDF Data by Location”.

This topic shows detailed information about “Record Trace - IFCID 001 - Destination Related Data”.

This topic shows detailed information about “Record Trace - IFCID 001 - Destination Related Data”.

This topic shows detailed information about “Record Trace - IFCID 001 - Global DDF Data”.

This topic shows detailed information about “Record Trace - IFCID 001 - IFCID Data”.

This topic shows detailed information about “Record Trace - IFCID 001 - Log Manager Data”.

This topic shows detailed information about “Record Trace - IFCID 001 - Subsystem Services Data”.

This topic shows detailed information about “Record Trace - IFCID 001 - QSST Data”.

This topic shows detailed information about “Record Trace - IFCID 001 - QVAS Data”.

This topic shows detailed information about “Record Trace - IFCID 001 - QVLS Data”.

This topic shows detailed information about “Record Trace - IFCID 001 - z/OS Metrics”.
IFCID 001 - Checkpoint and IFI Data

IFCID 001 - Checkpoint and IFI Data

This topic shows detailed information about “Record Trace - IFCID 001 - Checkpoint and IFI Data”.

Record trace - IFCID 001 - Checkpoint and IFI Data

The field labels shown in the following sample layout of “Record Trace - IFCID 001 - Checkpoint and IFI Data” are described in the following section.

CHECKPOINT AND IFI DATA

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHECKPOINT COUNT</td>
<td>The number of checkpoints DB2 has taken.</td>
</tr>
<tr>
<td></td>
<td>A checkpoint is a point at which DB2 records</td>
</tr>
<tr>
<td></td>
<td>internal status information to the DB2 log.</td>
</tr>
<tr>
<td></td>
<td>This information is used in the recovery</td>
</tr>
<tr>
<td></td>
<td>process if DB2 abends.</td>
</tr>
<tr>
<td>Background and Tuning Information</td>
<td></td>
</tr>
<tr>
<td></td>
<td>For Statistics reports only: A checkpoint is</td>
</tr>
<tr>
<td></td>
<td>taken when the specified number of log records</td>
</tr>
<tr>
<td></td>
<td>have been written. A checkpoint is also taken</td>
</tr>
<tr>
<td></td>
<td>each time DB2 switches to a new active log data</td>
</tr>
<tr>
<td></td>
<td>set. If the Statistics reporting period is</td>
</tr>
<tr>
<td></td>
<td>30 minutes and the value of this field is 15,</td>
</tr>
<tr>
<td></td>
<td>then DB2 is taking checkpoints every 2 minutes.</td>
</tr>
<tr>
<td></td>
<td>If the data sets are too small or the value for</td>
</tr>
<tr>
<td></td>
<td>LOGLOAD is too low, checkpoints occur too</td>
</tr>
<tr>
<td></td>
<td>frequently. As a result, database writes do not</td>
</tr>
<tr>
<td></td>
<td>perform efficiently. The frequency of DB2</td>
</tr>
<tr>
<td></td>
<td>checkpoints can be decreased by increasing the</td>
</tr>
<tr>
<td></td>
<td>value of the DSNZPARM LOGLOAD (CHECKPOINT FREQ</td>
</tr>
<tr>
<td></td>
<td>on the Tracing install panel).</td>
</tr>
<tr>
<td>Rule of thumb:</td>
<td>In a production environment, DB2 should take</td>
</tr>
<tr>
<td></td>
<td>checkpoints every 10 minutes or so.</td>
</tr>
<tr>
<td></td>
<td>The default value for LOGLOAD is 50000. The</td>
</tr>
<tr>
<td></td>
<td>actual value that you choose is dependent on</td>
</tr>
<tr>
<td></td>
<td>the volume and nature of the work performed by</td>
</tr>
<tr>
<td></td>
<td>your DB2 subsystem. It is a trade-off between</td>
</tr>
<tr>
<td></td>
<td>the performance efficiency of larger numbers</td>
</tr>
<tr>
<td></td>
<td>and the longer time to restart DB2 when there is</td>
</tr>
<tr>
<td></td>
<td>an abnormal termination.</td>
</tr>
<tr>
<td>Field Name:</td>
<td>QWSDCKPT</td>
</tr>
<tr>
<td></td>
<td>This is an exception field.</td>
</tr>
</tbody>
</table>

REASON STATISTICS INVOKED

The reason why statistics records were written.

Field Name: QWSDRINV

HIGH USED RBA

The high-used RBA address of the log (DB2 field prior to DB2 11: QWSDLR).

Field Name: QWSDLRG

IFI ABENDS
**IFI READA**
The number of calls made to IFI using the READA (read asynchronous data) function.

*Field Name:* QWSDSCRA

**DCAP.LOG REC.RETRIEVED**
The number of log records retrieved for which data capture processing was invoked.

*Field Name:* QWSDCDCDL

**DCAP.DATA ROWS RETURNED**
The total number of data capture data rows returned.

*Field Name:* QWSDCDCDDR

**IFI UNRECOG.**
The number of calls made to IFI using a function that is not recognized by the interface.

*Field Name:* QWSDSCCU

**IFI READS**
The number of calls made to IFI using the READS (read synchronous data) function.

*Field Name:* QWSDSCRS

**DCAP.LOG READS**
The total number of data capture log reads for processing IFI reads requests for IFCID 185.

*Field Name:* QWSDCDLR

**DCAP.DATA DESC.RETURNED**
The total number of data capture describes performed.

A data capture describe is the process of getting descriptive information about a DB2 table from the catalog.

*Field Name:* QWSDCDDD

**IFI COMMANDS**
The number of calls made to IFI using the COMMAND function.

*Field Name:* QWSDSCCO

**IFI WRITE**
The number of calls made to IFI using the WRITE function.

*Field Name:* QWSDSCWR

**DCAP.LOG REC.RETURNED**
The total number of data capture log records returned.

*Field Name:* QWSDCDRR
DCAP.DESCRIBES
The total number of data capture describes performed.
A data capture describe is the process of getting descriptive information about a DB2 table from the catalog.
Field Name: QWSDCDMB

DCAP.TABLES RETURNED
The total number of data capture tables returned to the caller of the IFI reads call for IFCID 185.
Field Name: QWSDCDTB

NO ROLLUP ACC RECS-ROLLUP THRESHOLD EXCEEDED
The number of roll-up accounting records written due to roll-up threshold exceeded.
Field Name: QWSDARTH

NO ROLLUP ACC RECS-ROLLUP STORAGE THRESHOLD EXC
The number of roll-up accounting records written due to roll-up accounting storage threshold exceeded.
Field Name: QWSDARSG

NO ROLLUP ACC RECS-STALENESS THRESHOLD EXCEEDED
The number of roll-up accounting records written due to staleness threshold exceeded.
Field Name: QWSDARST

NO RECS NOT QUALIFIED FOR ACC ROLLUP
The number of records that failed to qualify for accounting roll-up because all roll-up key fields are equal to NULL or because of NULL values that are not permitted.
Field Name: QWSDARIR
IFCID 001 - CPU Time Data

This topic shows detailed information about “Record Trace - IFCID 001 - CPU Time Data”.

This section shows CPU timer values for each resource manager and control address space.

Record trace - IFCID 001 - CPU Time Data

The field labels shown in the following sample layout of “Record Trace - IFCID 001 - CPU Time Data” are described in the following section.

CPU TIME DATA

PROCEDURE NAME
The last 4 characters of the procedure used to start the address space, or a constant identifier.

Field Name: QWSAPROC

ADDR SPACE ASID
The ASID of the address space.

Field Name: QWSAASID

ASCB
The ASCB token.

Field Name: QWSAASCB

SRB TIME
The accumulated SRB time for the address space. This value includes both, the preemptable and nonpreemptable SRB time. It does not include CPU time that is consumed on an IBM zIIP.

Field Name: QWSASRBT

PREEMP SRB TIME
The preemptible SRB timer value for the address space. This value does not include the CPU time that is consumed on an IBM zIIP.

Field Name: QWSAPSRRB

PREEMP SRB ZIIP
The preemptible SRB timer value that is consumed on an IBM zIIP for address space.

Field Name: QWSAPSRRB_ZIIP

TCB TIME
The accumulated job step time (TCB) for the address space.

Field Name: QWSAEJST

QWSAMCPU (S)
IFCID 001 - CPU Time Data

This field is for IBM service use.

Field Name: QWSAMCPU
ICFID 001 - DB2 Command Data

This topic shows detailed information about “Record Trace - IFCID 001 - DB2 Command Data”.

Record trace - IFCID 001 - DB2 Command Data

The field labels shown in the following sample layout of “Record Trace - IFCID 001 - DB2 Command Data” are described in the following section.

<table>
<thead>
<tr>
<th>DB2 Command Data</th>
<th>DISPLAY DB</th>
<th>0</th>
<th>DISPLAY THRD</th>
<th>0</th>
<th>DISP UTIL</th>
<th>0</th>
<th>DISP TRACE</th>
<th>0</th>
<th>DISPL RLIMIT</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>START DB</td>
<td>10</td>
<td>START TRACE</td>
<td>0</td>
<td>START DB2</td>
<td>0</td>
<td>START RLIM</td>
<td>0</td>
<td>STOP DB2</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>MODIFY TRACE</td>
<td>0</td>
<td>TERM UTILITY</td>
<td>0</td>
<td>START DDF</td>
<td>0</td>
<td>STOP DDF</td>
<td>0</td>
<td>CANCEL THREAD</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>DISPL LOCATN</td>
<td>0</td>
<td>UNREC CMDS</td>
<td>0</td>
<td>ARCH LOG</td>
<td>0</td>
<td>SET ARCH</td>
<td>0</td>
<td>DISPL ARCH</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RESET ENDOUT</td>
<td>0</td>
<td>ALTER BUFFER</td>
<td>0</td>
<td>DISP BUF</td>
<td>0</td>
<td>DISP GROUP</td>
<td>2</td>
<td>DISP PROCEDURE</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>RESET GENERIC</td>
<td>0</td>
<td>ALTER GBPOOL</td>
<td>0</td>
<td>DISP GBPOOL</td>
<td>0</td>
<td>START PROC</td>
<td>0</td>
<td>STOP PROCEDURE</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>DISPLAY GROUP</td>
<td>2</td>
<td>ALTER UTILITY</td>
<td>0</td>
<td>DISP FUNC</td>
<td>0</td>
<td>START FUNC</td>
<td>0</td>
<td>STOP FUNCTION</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>SET LOG</td>
<td>0</td>
<td>DISPLAY LOG</td>
<td>0</td>
<td>SET SYSPARM</td>
<td>0</td>
<td>DISPLAY DDF</td>
<td>2</td>
<td>ACCESS DB</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>START PROFILE</td>
<td>0</td>
<td>STOP PROFILE</td>
<td>0</td>
<td>DISP ACCEL</td>
<td>0</td>
<td>START ACCEL</td>
<td>0</td>
<td>STOP ACCEL</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

**DISPLAY DB**

The number of DB2 DISPLAY DATABASE commands issued to view objects within one or more DB2 databases. This includes normal and abnormal completion of the command.

*Field Name:* Q9STCTR0

**DISPLAY THRD**

The number of DB2 DISPLAY THREAD commands issued to view threads active within the DB2 subsystem. This includes normal and abnormal completion of the command.

*Field Name:* Q9STCTR1

**DISP UTIL**

The number of DB2 DISPLAY UTILITY commands issued to view the status of one or more DB2 utilities. This includes normal and abnormal completion of the command.

*Field Name:* Q9STCTR2

**DISP TRACE**

The number of DB2 DISPLAY TRACE commands issued to determine the currently active DB2 traces. This includes normal and abnormal completion of the command.

*Field Name:* Q9STCTRc

**DISPL RLIMIT**

The number of DB2 DISPLAY RLIMIT commands issued to view the current status of the DB2 resource limit facility. This includes normal and abnormal completion of the command.

*Field Name:* Q9STCTRg

**START DB**

The number of DB2 START DATABASE commands issued to make a database available for use. This includes normal and abnormal completion of the command.

*Field Name:* Q9STCTR5
The number of DB2 START TRACE commands issued to initiate a DB2 trace. This includes normal and abnormal completion of the command.

Field Name: Q9STCTR6

START DB2

The number of DB2 START DB2 commands issued to bring up a DB2 subsystem. This includes normal and abnormal completion of the command.

Field Name: Q9STCTR7

START RLIM

The number of DB2 START RLIMIT commands issued to enable the DB2 resource limit facility. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRE

STOP DB

The number of DB2 STOP DATABASE commands issued to prevent access to a DB2 database. This includes normal and abnormal completion of the command.

Field Name: Q9STCTR8

STOP TRACE

The number of DB2 STOP TRACE commands issued to terminate one or more active DB2 traces. This includes normal and abnormal completion of the command.

Field Name: Q9STCTR9

STOP DB2

The number of STOP DB2 commands. This includes both normal and abnormal completions.

Field Name: Q9STSCRA

STOP RLIM

The number of DB2 STOP RLIMIT commands issued to disable the DB2 resource limit facility. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRF

RECOV BSDS

The number of DB2 RECOVER BSDS commands issued to reestablish dual bootstrap data sets after one has been disabled by a data set error. This includes normal and abnormal completion of the command.

Field Name: Q9STCTR3

RECOV INDOUBT

The number of DB2 RECOVER INDOUBT commands issued to recover threads left indoubt because DB2 or a transaction manager could not automatically recover them. This includes normal and abnormal completion of the command.
Field Name: Q9STCTR4

MODIFY TRACE
The number of DB2 MODIFY TRACE commands issued to alter trace events (IFCIDs) for an active trace. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRL

UNREC CMDS
The number of commands not recognized by DB2. The number is incremented if the command verb or primary keyword cannot be determined. For example:
- "-DISPLOX DATABASE(*)" is an unknown verb.
- "-DISPLAY FATAFASE(*)" is an unknown primary keyword.

Field Name: Q9STCTRL

ARCH LOG
The number of DB2 ARCHIVE LOG commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRL

SET ARCH
IFCID 001 - DB2 Command Data

The number of DB2 SET ARCHIVE commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRLP

DISPL ARCH

The number of DB2 DISPLAY ARCHIVE commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRQ

RESET INDOUBT

The number of DB2 RESET INDOUBT commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTR7

ALTER BUFFER

The number of DB2 ALTER BUFFERPOOL commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTR9

DISP BUF

The number of DB2 DISPLAY BUFFERPOOL commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTR0

DISP GROUP

The number of DB2 DISPLAY GROUP commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTR8

DISP PROCEDURE

The number of DB2 DISPLAY PROCEDURE commands executed. This includes normal and abnormal completion of the command.

Field Name: Q9STCTR

RESET GENERIC

The number of DB2 RESET GENERICLU commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRD

ALTER GBPOOL

The number of DB2 ALTER GROUPBUFFERPOOL commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRS

DISP GBPOOL

The number of DB2 DISPLAY GROUPBUFFERPOOL commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRT

START PROC
The number of DB2 START PROCEDURE commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRV

STOP PROCEDURE
The number of DB2 STOP PROCEDURE commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRW

DISPLAY GROUP
The number of DB2 DISPLAY GROUP commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRX

ALTER UTILITY
The number of DB2 ALTER UTILITY commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRY

DISP FUNC
The number of DB2 DISPLAY FUNCTION commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRZ

START FUNC
The number of DB2 START FUNCTION commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTX0

STOP FUNCTION
The number of DB2 STOP FUNCTION commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTX1

SET LOG
The number of DB2 SET LOG commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTX2

DISPLAY LOG
The number of DB2 DISPLAY LOG commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTX3

SET SYSPARM
The number of DB2 SET SYSPARM commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTX4

DISPLAY DDF
IFCID 001 - DB2 Command Data

The number of DB2 DISPLAY DDF commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTX5

ACCESS DB

The number of DB2 ACCESS DATABASE commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTAD

START PROFILE

The number of DB2 START PROFILE commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTSS

STOP PROFILE

The number of DB2 STOP PROFILE commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTST

DISPLAY PROF

The number of DB2 DISPLAY PROFILE commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTSD

DISP PROFILE

The number of DB2 DISPLAY PROFILE commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTSD

DISP ACCEL

The number of DB2 DISPLAY ACCEL commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTDA

START ACCEL

The number of DB2 START ACCEL commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTSA

STOP ACCEL

The number of DB2 STOP ACCEL commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTXA

MODIFY DDF (DB2 V10 only)

The number of DB2 MODIFY DDF commands issued.

Field Name: Q9STCTMD
IFCID 001 - DDF Data by Location

This topic shows detailed information about “Record Trace - IFCID 001 - DDF Data by Location”.

Record trace - IFCID 001 - DDF Data by Location

The field labels shown in the following sample layout of “Record Trace - IFCID 001 - DDF Data by Location” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOCATION NAME (SHORT)</td>
<td>The name of the remote location.</td>
</tr>
<tr>
<td>LOCATION NAME (LONG)</td>
<td>The name of the remote location.</td>
</tr>
<tr>
<td>INITIATED CONVERSATIONS</td>
<td>The number of conversations that were initiated from the requester location.</td>
</tr>
<tr>
<td>MESSAGES SENT TO REMOTE</td>
<td>The number of messages sent to the remote location.</td>
</tr>
<tr>
<td>SQL STMS SENT TO REMOTE</td>
<td></td>
</tr>
<tr>
<td>BYTES SENT TO REMOTE</td>
<td></td>
</tr>
<tr>
<td>ROWS SENT TO REMOTE</td>
<td></td>
</tr>
<tr>
<td>COMMIT REQUESTS SENT</td>
<td></td>
</tr>
<tr>
<td>ABORT REQUESTS SENT</td>
<td></td>
</tr>
<tr>
<td>INDOUBT THREADS</td>
<td></td>
</tr>
<tr>
<td>BACKOUT REQS SENT TO PART</td>
<td></td>
</tr>
<tr>
<td>BACKOUT REQS RECV FR COORD</td>
<td></td>
</tr>
<tr>
<td>SWITCH TO LIMITED BLOCK MODE</td>
<td></td>
</tr>
<tr>
<td>COMMIT REQS RECV FR COORD</td>
<td></td>
</tr>
<tr>
<td>COMMIT REQS SENT TO PART</td>
<td></td>
</tr>
<tr>
<td>LAST AGNT REQS RECV FR INIT</td>
<td></td>
</tr>
<tr>
<td>PREPARE REQS RECV FR COORD</td>
<td></td>
</tr>
<tr>
<td>ROLLBACKS PERFORMED</td>
<td></td>
</tr>
<tr>
<td>FORGET RESP RECV FR PART</td>
<td></td>
</tr>
<tr>
<td>THREAD ALLOC REQS RECEIVED</td>
<td></td>
</tr>
<tr>
<td>COMMIT REQS RECV FR PARTIC</td>
<td></td>
</tr>
</tbody>
</table>

LOCATION NAME (SHORT)

The name of the remote location.

Field Name: QLSTLOCN

LOCATION NAME (LONG)

The name of the remote location.

Field Name: QLSTLOCN

INITIATED CONVERSATIONS

The number of conversations that were initiated from the requester location. This value is maintained at the requester location.

A conversation is a specific instance of using TCP/IP or SNA LU 6.2 to transfer information between a requester and a server. A conversation is a logical connection between a requester and a server.

Field Name: QLSTCNVS

INITIATED FROM REMOTE SITE

The number of conversations that were initiated from the requester to the server location. This value is updated at the server location.

Field Name: QLSTCNVR

MESSAGES SENT TO REMOTE

The number of messages sent to the remote location. A message is a group of characters and control bit sequences transferred on a single TCP/IP or SNA API call. This value is maintained at the location where the messages originated.
Field Name: QLSTMSGS

SQL STMTS SENT TO REMOTE
The number of SQL statements sent to the remote server. This value is updated at the requester location.

Field Name: QLSTSQLS

BYTES SENT TO REMOTE
The number of bytes of data sent to the requester location. This value is maintained at the server location.

Field Name: QLSTBYTS

ROWS SENT TO REMOTE
The number of data rows sent to the requester location (includes SQLDA). This value is updated at the server location.

Field Name: QLSTROWS

BLOCKS TRANSMITTED
The number of blocks transmitted using block fetch. This value is maintained at the server location.

Field Name: QLSTBTBF

COMMIT REQUESTS SENT
This field depends on the DB2 version that is installed:

- DB2 10 or later: The number of commit requests sent to the server (single-phase commit protocol) and the committed requests sent to the participant (two-phase commit protocol).
- Prior to DB2 10: The number of commit requests sent to the server location (single-phase commit operations only). This value is maintained at the requester location.

Field Name: QLSTCOMS

INDOUBT THREADS
The number of threads that became indoubt with the remote location as the coordinator (two-phase commit operations only). A large value might indicate network problems.

Field Name: QLSTINDT

BACKOUT REQS SENT TO PART
The number of backout requests sent to the participant (two-phase commit operations only).

Field Name: QLSTBKSE

ROWS IN THE MESSAGE BUFFER
The number of rows transmitted or received in DB2 message buffers using block fetch. This field includes both requester and server activity.

Field Name: QLSTBROW

This is an exception field.

COMMTIS WITH REMOTE COORD
The number of commit operations performed with the remote location as the coordinator (two-phase commit operations only).

**Field Name:** QLSTCPTR

This is an *exception* field.

**COMMIT REQS RECV FR COORD**

The number of commit requests received from the coordinator (two-phase commit operations only).

**Field Name:** QLSTCRRC

This is an *exception* field.

**LAST AGNT REQS RECV FR INIT**

The number of last agent requests received from the initiator (two-phase commit operations only).

This number is incremented when the DB2 server is receiving a last agent request from its requester.

**Field Name:** QLSTLARC

This is an *exception* field.

**PREPARE REQS RECV FR COORD**

The number of prepare requests received from the coordinator (two-phase commit operations only).

**Field Name:** QLSTPRRC

This is an *exception* field.

**SQL STMTS BOUND FOR REM ACC**

The number of SQL statements that were bound for remote access (DB2 private protocol only). This value is maintained at the requester location.

**Field Name:** QLSTRBND

This is an *exception* field.

**FORGET RESP RECV FR PART**

The number of forget responses received from the participant (two-phase commit operations only). This indicates that the participant was read-only.

**Field Name:** QLSTRRRRC

This is an *exception* field.

**THREAD ALLOC REQS RECEIVED**

The number of DBAT allocation requests received from the remote location. This value is only meaningful at the server location.

**Field Name:** QLSTTRNR

This is an *exception* field.

**BACKOUT RESP RECV FR PARTIC**

The number of backout responses received from the participant (two-phase commit operations only). This indicates that the participant voted no to the prepare request.

**Field Name:** QLSTVNRC
IFCID 001 - DDF Data by Location

This is an exception field.

COMMIT RESP RECV FR PARTIC
The number of request commit responses received from the participant (two-phase commit operations only).
Field Name: QLSTVYRC
This is an exception field.

PRDID REMOTE LOCATION
The product ID and version of the remote location.
Field Name: QLSTPRID

DEALLOCATED CONVERSATIONS
The number of conversations that were deallocated from this site to the remote site.
Field Name: QLSTCNVT

MESSAGES RECV FR REMOTE
The number of messages received by VTAM from the remote location. This value is maintained at the location where the messages were received.
More messages might be sent from the server location than are received by the requester due to the manner in which distributed SQL statements are processed internally.
Field Name: QLSTMSGR

SQL STMTS RECV FR REMOTE
The number of SQL statements received from the requester location. This value is updated at the server location.
Field Name: QLSTSQLR

BYTES RECV FR REMOTE
The number of bytes of data received from the server location. This value is maintained at the requester location.
More bytes of data might be sent from the server location than are received by the requester due to the manner in which distributed SQL statements are processed internally.
Field Name: QLSTBYTR

ROWS RETRIEVED FR REMOTE
The number of data rows received from the server location. This value is maintained at the requester location.

Note:
• This value does not include any SQLDA or SQLCA transmitted.
• Block fetch can significantly affect the number of rows sent across the network. When used with nonupdate cursors, block fetch groups as many rows as possible into the message buffer, and transmits the buffer over the network without requiring a VTAM message. Consequently, more rows of data might be sent from the server location than are received by the requester location.
This is especially true when DB2 private protocol is used because multiple blocks can be transmitted from the server with no intervening messages from the requester.

**Field Name: QLSTROWR**

**BLOCKS RECEIVED**

The number of blocks received from the remote location using block fetch. This value is maintained at the requester location.

**Field Name: QLSTBRBF**

**COMMIT REQUESTS RECEIVED**

This field depends on the DB2 version that is installed:

- **DB2 10 or later**: The number of commit requests received from the requester (single-phase commit protocol) and committed requests received from the coordinator (two-phase commit protocol).
- **Prior to DB2 10**: The number of commit requests received from the requester location (single-phase commit operations only). This value is maintained at the server location.

**Field Name: QLSTCOMR**

**ABORT REQUESTS SENT**

This field depends on the DB2 version that is installed:

- **DB2 10 or later**: The number of abort requests sent to the server (single-phase commit protocol) and backout requests sent to the participant (two-phase commit protocol).
- **Prior to DB2 10**: The number of rollback requests sent to the remote server location (single-phase commit operations only). This value is maintained at the requester location.

**Field Name: QLSTABRS**

**ABORT REQUESTS RECEIVED**

This field depends on the DB2 version that is installed:

- **DB2 10 or later**: The number of abort requests received from the requester (single-phase commit protocol) and backout requests received from the coordinator (two-phase commit protocol).
- **Prior to DB2 10**: The number of rollback requests received from the requester location (single-phase commit operations only). This value is maintained at the server location.

**Field Name: QLSTABRR**

**CONV REQUESTS QUEUED**

The number of conversation requests queued by the distributed data facility and waiting for allocation. This value is maintained at the requester location.

**Background and Tuning Information**

When this value is high, increase the limit for the number of conversations.

**Field Name: QLSTCNVQ**

This is an exception field.

**BACKOUT REQS RECV FR COORD**
The number of backout requests received from the coordinator (two-phase commit operations only).

**Field Name:** QLSTBKRC

This is an exception field.

**SWITCH TO LIMITED BLCK MODE**

The number of times the continuous block fetch was switched to a limited block fetch (DB2 private protocol only). This value is maintained at the requester location.

**Background and Tuning Information**

When this value is high, consider tuning VTAM.

**Field Name:** QLSTCBLB

This is an exception field.

**COMMIT REQS SENT TO PART**

The number of commit requests sent to the participant (two-phase commit operations only).

**Field Name:** QLSTCRSE

This is an exception field.

**LAST AGNT REQS SENT TO COOR**

The number of last agent requests sent to the coordinator (two-phase commit operations only).

A last agent request reduces the number of messages to be sent for the commit. When DB2 is the requester, this number is incremented when a conversation is deallocated and the conversation was not used since the last commit.

**Background and Tuning Information**

If this number is large and your application design allows for it, you can store another message by issuing a release before the commit (only for a DB2 requester).

**Field Name:** QLSTLASE

**PREPARE REQS SENT TO PART**

The number of prepare requests sent to the participant (two-phase commit operations only).

**Field Name:** QLSTPRSE

**ROLLBACKS PERFORMED**

The number of rollback operations performed with the remote location as the coordinator (two-phase commit operations only).

**Field Name:** QLSTRBTR

**FORGET RESP SENT TO COORD**

The number of forget responses sent to the coordinator (two-phase commit operations only). This indicates that the participant was read-only.

**Field Name:** QLSTRRSE

**THREAD ALLOC REQS SENT**
IFCID 001 - DDF Data by Location

The number of DBAT allocation requests sent to the remote location. This value is only meaningful at the requester location.

Field Name: QLSTTRNS

BACKOUT RESP SENT TO COORDI

The number of backout responses sent to the coordinator (two-phase commit operations only). This indicates that the participant voted no to the prepare request.

Field Name: QLSTVNSE

COMMIT RESP SENT TO COORDI

The number of request commit responses sent to the coordinator (two-phase commit operations only).

Field Name: QLSTVYSE
IFCID 001 - Destination Related Data

IFCID 001 - Destination Related Data
This topic shows detailed information about “Record Trace - IFCID 001 - Destination Related Data”.

This record contains one data section for each destination.

Record trace - IFCID 001 - Destination Related Data
The field labels shown in the following sample layout of “Record Trace - IFCID 001 - Destination Related Data” are described in the following section.

<table>
<thead>
<tr>
<th>DESTINATION RELATED DATA</th>
<th>DEST NAME</th>
<th>SMF</th>
<th>SEQNO</th>
<th>310 RECS WRITTEN</th>
<th>310 RECS NOT WRITTEN</th>
<th>0 BUFFER ERRORS</th>
<th>0 NOT ACTIVE ERRORS</th>
<th>0 RECS NOT ACCEPTED</th>
<th>0 WRITER FAILURES</th>
<th>QWSBOTH1</th>
<th>QWSBOTH2</th>
<th>QWSBOTH3</th>
<th>QWSBOTH4</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEST NAME</td>
<td>GTF</td>
<td>Generalized trace facility</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SMF</td>
<td>System management facilities</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RES</td>
<td>Resident trace table (not accumulated)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>OPN</td>
<td>Special destination for IFI READA buffered records</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>All other values are shown in hexadecimal.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Field Name: QWSBNM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

SEQNO
The unique destination sequence of the last record written to the destination.
Field Name: QWSBWSEQ

RECS WRITTEN
The number of records written to the destination.
Field Name: QWSBSRSW

RECS NOT WRITTEN
The number of records not written to the destination.
Field Name: QWSBSRNW

BUFFER ERRORS
The number of SMF buffer-overrun errors.
Field Name: QWSBSBUF

NOT ACTIVE ERRORS
The number of times SMF was not active.
Field Name: QWSBSACT

RECS NOT ACCEPTED
The number of records not accepted by the destination writer.
Field Name: QWSBSRNA

WRITER FAILURES
IFCID 001 - Destination Related Data

The number of write failures to the destination.

Field Name: QWSBSWF
IFCID 001 - Global DDF Data

Record trace - IFCID 001 - Global DDF Data

The field labels shown in the following sample layout of “Record Trace - IFCID 001 - Global DDF Data” are described in the following section.

**GLOBAL DDF DATA**

```
<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBAT/CONN QUEUED-MAX ACTIVE</td>
<td>The number of times a DBAT or connection was queued because it reached the ZPARM maximum for active remote threads (MAXDBAT).</td>
</tr>
<tr>
<td>CONN CLOSED - MAX QUEUED</td>
<td></td>
</tr>
<tr>
<td>COLD START CONNECTIONS</td>
<td></td>
</tr>
<tr>
<td>RESYNCHRONIZATION ATTEMPTED</td>
<td></td>
</tr>
<tr>
<td>CUR TYPE 1 INACTIVE DBATS</td>
<td></td>
</tr>
<tr>
<td>TYPE 1 CONNECTIONS TERMINAT</td>
<td></td>
</tr>
<tr>
<td>CUR QU INACT CONNS (TYPE 2)</td>
<td></td>
</tr>
<tr>
<td>CUR QU INACT CONNS (TYPE 2)</td>
<td></td>
</tr>
<tr>
<td>MIN QUEUE TIME</td>
<td></td>
</tr>
<tr>
<td>MAX QUEUE TIME</td>
<td></td>
</tr>
<tr>
<td>AVG QUEUE TIME</td>
<td></td>
</tr>
<tr>
<td>CUR ACTIVE AND DISCON DBATS</td>
<td></td>
</tr>
<tr>
<td>HWM TOTAL REMOTE CONNECTIONS</td>
<td></td>
</tr>
<tr>
<td>CUR DISCON DBATS NOT IN USE</td>
<td></td>
</tr>
<tr>
<td>DBATS CREATED</td>
<td></td>
</tr>
<tr>
<td>CUR DBATS-BND DEALLC</td>
<td></td>
</tr>
</tbody>
</table>
```

**DBAT/CONN QUEUED-MAX ACTIVE**

The number of times a DBAT or connection was queued because it reached the ZPARM maximum for active remote threads (MAXDBAT).

**Field Name:** QDSTQDBT

This is an exception field.

**CONN REJECTED-MAX CONNECTED**

The number of connections that were rejected because the ZPARM limit for maximum remote connections (CONDBAT) was reached.

**Field Name:** QDSTQCRT

**CONN CLOSED - MAX QUEUED**

The number of queued client connections whose TCP/IP sockets were closed because the system parameter MAXCONQN was exceeded.

The socket close only occurs when the DB2 subsystem is a member of a data sharing group and DB2 was started with DDF THREADS set to INACTIVE.

**Field Name:** QDSTNCQC

**QUEUED CLIENT CONNECTIONS**

The number of queued client connections whose TCP/IP socket were closed due to system parameter MAXCONQW being exceeded.

The socket close only occurs when the DB2 subsystem is a member of a data sharing group and DB2 was started with DDF THREADS set to INACTIVE.

**Field Name:** QDSTNCWC

**COLD START CONNECTIONS**
The number of cold start connections with all remote locations (two-phase commit operations only).

Field Name: QDSTCSTR
This is an exception field.

WARM START CONNECTIONS
The number of warm start connections with all remote locations (two-phase commit operations only).

Field Name: QDSTWSTR
This is an exception field.

RESYNCHRONIZATION ATTEMPTED
The number of resynchronization connections attempted with all remote locations (two-phase commit operations only).

Background and Tuning Information
A large value can indicate network or system problems.

Field Name: QDSTRSAT
This is an exception field.

RESYNCHRONIZATION SUCCEEDED
The number of resynchronization connections that succeeded with all remote locations (two-phase commit operations only).

Background and Tuning Information
If the value of this field is much less than the number of resynchronizations attempted, network problems might exist.

Field Name: QDSTRSSU
This is an exception field.

CUR TYPE 1 INACTIVE DBATS
The current number of inactive DBATs type 1 (snapshot).

Field Name: QDSTQCIT

HWM TYPE 1 INACTIVE DBATS
The maximum number of inactive type 1 DBATs.
This value is a high-water mark.

Field Name: QDSTQMIT
This is an exception field.

TYPE 1 CONNECTIONS TERMINATED
The number of threads or connections that were terminated instead of being made type 1 inactive because the maximum number of type 1 inactive threads was reached (MAXTYPE1).

Field Name: QDSTNITC

CUR INACTIVE CONNS (TYPE 2)
The current number of type 2 inactive connections.

Field Name: QDSTCIN2
IFCID 001 - Global DDF Data

HWM INACTIVE CONNS (TYPE 2)
The maximum number of concurrent type 2 inactive connections that existed.
This value is a high-water mark for QDSTCIN2.
Field Name: QDSTMIN2

ACC QU INACT CONNS (TYPE 2)
The number of RECEIVE requests on type 2 inactive or new connections that are queued to be serviced by a disconnected (pooled) DBAT.
Field Name: QDSTQIN2

CUR QU INACT CONNS (TYPE 2)
The current number of type 2 inactive or new connections that are queued waiting for a database access thread (DBAT).
Field Name: QDSTNQR2

MIN QUEUE TIME
The minimum queue time of a type 2 inactive or new connection that was queued waiting for a database access thread (DBAT) in the last statistical period.
Field Name: QDSTNQMNN

MAX QUEUE TIME
The maximum queue time of a type 2 inactive or new connection that was queued waiting for a database access thread (DBAT) in the last statistical period.
Field Name: QDSTNQMX

AVG QUEUE TIME
The average queue time of a type 2 inactive or new connection that was queued waiting for a database access thread (DBAT) in the last statistical period.
Field Name: QDSTNQAV

HWM QU INACT CONNS (TYPE 2)
The maximum number of type 2 inactive or new connections that are queued waiting for a database access thread.
This value is a high-water mark for QDSTNQR2.
Field Name: QDSTMQR2

CUR ACTIVE AND DISCON DBATS
The current number of active and disconnected (pooled) DBATs.
Field Name: QDSTCNAT

HWM ACTIVE AND DISCON DBATS
The maximum number of active and disconnected (pooled) DBATs that existed.
This value is a high-water mark for QDSTCNAT.
Field Name: QDSTHWAT
This is an exception field.

**HWM TOTL REMOTE CONNECTIONS**

The maximum number of active and remote connections. This value is a high-water mark.

**Field Name:** QDSTHWDT

This is an exception field.

**CUR DISCON DBATS NOT IN USE**

The current number of disconnected (pooled) DBATs that are available to process type 2 inactive or new connections.

**Field Name:** QDSTNADS

**HWM DISCON DBATS NOT IN USE**

The maximum number of disconnected (pooled) DBATs that are available to process type 2 inactive or new connections.

This value is a high-water mark for QDSTNADS.

**Field Name:** QDSTMADS

**DBATS CREATED**

The number of requests that required a database access thread (DBAT) to be created to process the request.

**Note:** This does not include database access threads created to replace disconnected (pooled) DBATs that terminated because they reached their reuse limit.

**Field Name:** QDSTNDBA

**DISCON (POOL) DBATS REUSED**

The number of requests that were satisfied by assigning a disconnected (pooled) DBAT to process the request.

**Field Name:** QDSTPOOL

**CUR ACTIVE DBATS-BND DEALLC**

The current number of DBATs that are active because the associated packages were bound with RELEASE(DEALLOCATE).

**Field Name:** QDSTNARD

**HWM ACTIVE DBATS-BND DEALLC**

The maximum number of DBATs that are active because the associated packages were bound with RELEASE(DEALLOCATE).

**Field Name:** QDSTMARD
IFCID 001 - IFCID Data

IFCID 001 - IFCID Data
This topic shows detailed information about “Record Trace - IFCID 001 - IFCID Data”.

This record contains one data section for each active IFCID.

Record trace - IFCID 001 - IFCID Data
The field labels shown in the following sample layout of “Record Trace - IFCID 001 - IFCID Data” are described in the following section.

IFCID DATA

IFCID  1  IFCID SEQNO  16  RECS WRITTEN  15  RECS NOT WRITTEN  15  RECS NOT DESIRED  0
BUFFER NOT AVAILABLE  0  COLLECT FAILURES  0
QWSCOTH1  0  QWSCOTH2  0

IFCID
The IFCID for the following statistics.
Field Name: QWSCIID

IFCID SEQNO
The last sequence number used for this IFCID.
Field Name: QWSCISEQ

RECS WRITTEN
The number of records successfully written for this IFCID.
Field Name: QWSCSRSW

RECS NOT WRITTEN
The number of records not written to this IFCID.
Field Name: QWSCSRNW

RECS NOT DESIRED
The number of records not desired.
Field Name: QWSCSRND

BUFFER NOT AVAILABLE
The number of errors due to the buffer not being available.
Field Name: QWSCSBNA

COLLECT FAILURES
The number of collection failures.
Field Name: QWSCSCF
IFCID 001 - Log Manager Data

This topic shows detailed information about “Record Trace - IFCID 001 - Log Manager Data”.

Record trace - IFCID 001 - Log Manager Data

The field labels shown in the following sample layout of “Record Trace - IFCID 001 - Log Manager Data” are described in the following section.

WRITE REQUEST-WAIT

The number of wait log write requests. Wait indicates that the log record is first written to the log buffer and then to the log data set.

Field Name: QJSTWRW

READ FROM OUTPUT BUFFER

The number of log reads satisfied from the output buffer.

Background and Tuning Information

This field, together with the reads satisfied from active log and reads satisfied from archive log (QJSTRACT and QJSTRARH) fields indicate how efficiently DB2 retrieves log records. Use these numbers to adjust the number of output buffers and the total active log capacity to maximize DB2 performance.

Field Name: QJSTRBUF

This is an exception field.

WRITE REQUEST-NO WAIT

The number of log write requests.

The log record is written asynchronously to the log buffer. The application does not wait for the record to be written to the log data set and regains control immediately.

Buffered log records are written to DASD when the buffer threshold is exceeded.

Field Name: QJSTWRNW

This is an exception field.

READ FROM ACTIVE LOG

The number of log reads satisfied from the active log data set.

Background and Tuning Information

This field, together with the reads satisfied from archive log and reads satisfied from output buffer fields, indicate how efficiently DB2 retrieves
log records. Use these numbers to adjust the number of output buffers and the total active log capacity to maximize DB2 performance. Ideally, this value should be 0 or very small.

Field Name: QJSTRACT
This is an exception field.

WRITE REQUEST-FORCE
The number of force log write requests. Force indicates that the log record is written to the log buffer, forcing the buffer to be written to the log data set on DASD.

Field Name: QJSTWRF

READ FROM ARCHIVE LOG
The number of log reads satisfied from archive log data sets.

Field Name: QJSTRARY

WRITE LOG BUFFER
The number of calls to the log write routine. This does not represent the number of physical log I/Os.

Field Name: QJSTBFWR

READ DELAY-TAPE VOLUME CONTENTION
The number of read accesses that were delayed because of a tape volume contention when only one reader per tape is possible.

Background and Tuning Information
This field shows the number of agents forced to wait because a tape volume was already in use by another. If this number is not 0, increase the read tape units on the archive log data set parameters panel DSNTIPA.

Field Name: QJSTTVC
This is an exception field.

WRITE I/O REQUESTS
The total number of log-write I/O requests (such as media manager calls). This is the sum of the IFCID 038/039 pairs and includes both copy1 and copy2 active log data set writes.

Background and Tuning Information
This value should correspond to the active log write I/O activity in an RMF report.

Field Name: QJSTLOGW

READ DELAY-UNAVAILABLE RESOURCE
The number of read accesses delayed due to unavailable resources.

Background and Tuning Information
Generally, this can be due to insufficient tape units allocated. If this is so, reissue the SET ARCHIVE command and use a higher value for the count parameter. Another (although unlikely) cause is insufficient archive log read service task availability.

Field Name: QJSTWUR
This is an *exception* field.

**WRITE BUFFER SCHEDULED-THRESHOLD**

The number asynchronous log write requests made because the log write threshold was reached.

**Background and Tuning Information**

This counter is provided primarily for an internal check. It is recommended to use the default write threshold of 20 buffers.

*Field Name:* QJSTTHRW

**CI CREATED-ACTIVE LOG**

The number of active log output control intervals created.

**Background and Tuning Information**

Log records are placed sequentially in output log buffers, which are formatted as VSAM control intervals. The control intervals are written to a set of predefined DASD active log data sets, which are used sequentially and recycled.

The ratio of this field to write output log buffers should be low.

**Rules of thumb:**

The lower the value, the better. A high value indicates that too many I/Os are required for the number of log buffers created.

It is possible that WRTTHRSN is set too low. It is also possible that transactions could be arriving so infrequently that at commit time force requests are not queued and each force request is individually triggering an I/O of its log buffers.

*Field Name:* QJSTBFFL

**WRITE BUFFER PAGED IN**

The number of times an output log buffer had to be paged in before it could be initialized. The log-write latch is held at this point.

**Background and Tuning Information**

A nonzero value could indicate that the output log buffer size is too large, or there is insufficient real storage to back up the output log buffer size.

*Field Name:* QJSTBPAG

**CI OFFLOADED**

The number of control intervals (CIs) offloaded from the active log to the archive log.

*Field Name:* QJSTCIOF

**WAIT FOR UNAVAILABLE LOG BUFFER**

The number of waits caused by an unavailable output log buffer.

When DB2 wants to write a log record and the log buffer is not available, DB2 and the application must wait for an available log buffer.

**Background and Tuning Information**

Another possible cause is that the size of the write threshold might be too close to the size of the output buffer.
If this field is not 0, increase the number in the output buffer field on installation panel DSNTIPL to increase the number of output buffers or increase the size of the buffer.

Field Name: QJSTWTB
This is an exception field.

CI WRITTEN
The total number of log control intervals (CIs) written. This includes CI rewrites and both copy1 and copy2 active log data set writes. If a given CI is rewritten 5 times, this counter is incremented by 5.

Field Name: QJSTCIWR

TOTAL BSDS ACCESS REQUESTS
The number of BSDS access requests.

Field Name: QJSTBSDS

CI SERIAL WRITE
The number of serial log write I/O requests. A serial log write I/O request occurs when DB2 rewrites a log CI that was previously written as a partial CI, in a dual logging environment. This value includes COPY1 and COPY2 active log data set writes. The difference between WRITE I/O REQ and CI SERIAL WRITE gives the number of parallel log write I/O requests. Typically, the first CI in a list of one start I/O is written serially, and the remaining to both COPY1 and COPY2 active log data sets. This value is meaningful only when DB2 runs in dual active log mode.

Field Name: QJSTSERW

ARCHIVE READ ALLOCATIONS
The number of archive log read allocations. It indicates the frequency of archive log open and close activity.

Background and Tuning Information
A high number indicates a need for more or larger active log data sets. This value should be small, ideally 0.

Field Name: QJSTALR

LOOK-AHEAD TAPE VOL MOUNTS ATTEMPTED
The number of look ahead (tape volume) mounts attempted.

Background and Tuning Information
This field and field QJSTLAMS (label LOOK-AHEAD MOUNT SUCCESSFUL) show the efficiency of look ahead for tape mounts.

Field Name: QJSTLAMA

ARCHIVE WRITE ALLOCATIONS
The number of archive log write allocations. It indicates the frequency of archive log open and close activity.

Background and Tuning Information
A high number indicates a need for more or larger active log data sets. This value should be small, ideally 0.
Field Name: QJSTALW
LOOK-AHEAD TAPE VOL MOUNTS SUCCEEDED
The number of successful look-ahead (tape volume) mounts. It indicates the look-ahead mounting performance gains.

Background and Tuning Information
For maximum performance, this field and field QJSTLAMA (label LOOK-AHEAD MOUNT ATTEMPTED) should be equal. To find the number of failed attempts, subtract the value in this field from LOOK-AHEAD MOUNT ATTEMPTED. Too many failed attempts negate potential performance gains. This can be caused by not having enough tape units available. Issue the DISPLAY ARCHIVE command and note the current count value. Then issue the SET ARCHIVE command using a higher value for the count parameter.

Field Name: QJSTLAMS

QJSTLSUS
This field is for IBM service use.

Field Name: QJSTLSUS

QJSTSPNN
This field is for IBM service.

Field Name: QJSTSPNN

QJSTSPNI
This field is for IBM service.

Field Name: QJSTSPNI

QJSTCLID
This field is for IBM service.

Field Name: QJSTCLID

QJSTCL2
This field is for IBM service.

Field Name: QJSTCL2

QJSTCLSN
This field is for IBM service.

Field Name: QJSTCLSN

QJSTAVAL
This field is for IBM service.

Field Name: QJSTAVAL
IFCID 001 - Subsystem Services Data

This topic shows detailed information about “Record Trace - IFCID 001 - Subsystem Services Data”.

Record trace - IFCID 001 - Subsystem Services Data

The field labels shown in the following sample layout of “Record Trace - IFCID 001 - Subsystem Services Data” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>IDENTIFY</td>
<td>66</td>
</tr>
<tr>
<td>CREATE THREAD</td>
<td>155</td>
</tr>
<tr>
<td>UR INDOUBT</td>
<td>0</td>
</tr>
<tr>
<td>COMMIT PH 2</td>
<td>0</td>
</tr>
<tr>
<td>ROLLBACK</td>
<td>0</td>
</tr>
<tr>
<td>SIGNON</td>
<td>0</td>
</tr>
<tr>
<td>UR INDOUBT RESOLV</td>
<td>0</td>
</tr>
<tr>
<td>COMMIT PH 1</td>
<td>0</td>
</tr>
<tr>
<td>SSAM EOM</td>
<td>0</td>
</tr>
<tr>
<td>TERMIN.THREAD</td>
<td>221</td>
</tr>
<tr>
<td>EXITS</td>
<td>66</td>
</tr>
<tr>
<td>SYNCRS</td>
<td>164</td>
</tr>
<tr>
<td>SSAM EOT</td>
<td>1</td>
</tr>
<tr>
<td>CRT.THRD QUID</td>
<td>0</td>
</tr>
<tr>
<td>SUBS.INT.CALLS</td>
<td>77</td>
</tr>
<tr>
<td>READ ONLY COMMIT</td>
<td>0</td>
</tr>
<tr>
<td>IDBACK*</td>
<td>1</td>
</tr>
<tr>
<td>IDFORE*</td>
<td>2</td>
</tr>
<tr>
<td>CTHREAD*</td>
<td>2</td>
</tr>
</tbody>
</table>

* = HIGH WATER MARK

IDENTIFY

The number of successful connections to DB2 from an allied address space (TSO, BATCH, CICS, IMS, CAF, or UTILITY).

Field Name: Q3STIDEN

CREATE THREAD

The number of successful create thread requests. It does not include DBATs.

A thread is required before an application can use SQL. When established, a thread can have one or more secondary authorization IDs.

A thread is needed to perform any DB2 activity. For example, a thread is needed to run a DB2 utility to perform an IFI request such as READS, or to process a DB2 command such as -DISPLAY THREAD. However, a thread is not created if the command failed because of a syntax error.

Background and Tuning Information

Thread reuse can help improve performance.

The term thread reuse only applies to IMS and CICS attachments. In the case of the TSO attachment facility and the call attachment facility (CAF), threads cannot be reused, because the threads are allocated to the user address space.

Thread reuse should be considered in the following cases:

• If transaction volume is high:
   High volume transactions should achieve a high percentage of thread reuse. If threads are reused on low volume transactions, the number of threads needed increases because these threads are not automatically terminated by IMS when not being used. This may result in too many idle threads for the level of the DB2 workload. Under CICS, protected threads are terminated after about 45 seconds if no transaction eligible to reuse the thread has been received.

• If thread creation cost is significant:
   As a rule of thumb, more than 5% of the total CPU cost of transaction processing is considered significant.

The ACQUIRE and RELEASE parameters of BIND should be specified to minimize the thread creation cost, while providing the needed concurrency:
If most of the application plan's SQL statements are executed, then ACQUIRE(ALLOCATE) is cheaper than ACQUIRE(USE).

If only a small number of the SQL statements are executed, ACQUIRE(USE) becomes cheaper and improves concurrency, because the required resources are only acquired (locked) when the plan actually references (uses) them. An example would be a generalized plan used by many different transactions. It would contain multiple logic paths referencing different tables.

Note that, when packages are involved, ACQUIRE(USE) is always implicitly used.

Concurrency in thread reuse is based on page locking provided by the IS and IX intent locks, whose duration is governed by ACQUIRE and RELEASE of BIND.

RELEASE(DEALLOCATE) is strongly recommended for thread-reuse transactions to reduce transaction CPU time.

When thread reuse is implemented, monitor the EDM pool. It should be sufficient in size to accommodate expanding plans where the next transaction requires additional plan sections over those that are already part of the plan.

Field Name: Q3STCTHD

This is an exception field.

UR INDOUBT

The number of indoubt units of recovery.

A unit of recovery is indoubt when a failure occurs after a successful prepare but before a successful commit. The failure can occur in the address space of the application, the transaction manager, DB2, or all of these. IMS and CICS applications use the prepare and commit sequence to commit work. Ideally, this value should be 0.

Field Name: Q3STINDT

This is an exception field.

COMMIT PH 2

The number of successful commit phase 2 in a two-phase environment such as CICS or IMS. It includes successfully committed agents associated with threads that use the Recoverable Resource Manager Services Attach Facility (RRSAF). It does not include successful single-phase commits or distributed two-phase commits.

Background and Tuning Information

IMS and CICS applications use the PREPARE and COMMIT sequence to commit work. A nonzero value for this field indicates that updates have occurred.

Field Name: Q3STCOMM

ROLLBACK

The number of times a unit of recovery was successfully rolled back. Some reasons for a rollback include:

- Application program abend
- Application rollback request
- Application deadlock on database records
IFCID 001 - Subsystem Services Data

- Application canceled by operator
- Thread abend due to resource shortage

This number also includes successfully aborted agents associated with threads that use the Recoverable Resource Manager Services Attach Facility (RRSAF).

Field Name: Q3STABRT
This is an exception field.

SIGNON
The number of signons that identified a new user of an existing thread for IMS and CICS.

This field is valid only for CICS and IMS (not valid for TSO, CAF, or UTILITY).

The initial signon does not perform an authorization check because the thread does not exist yet, but a resignon can.

Background and Tuning Information
If the number of signons is greater than the number of create thread occurrences, some threads have been reused. In the case of the TSO attachment facility and the call attachment facility (CAF), there is no sign-on, because the user is identified when the TSO address space is connected.

Field Name: Q3STSIGN
This is an exception field.

UR INDOUBT RESOLV
The number of indoubt units of recovery successfully resolved, either automatically or manually. It includes successful indoubt resolutions for agents associated with threads that use the Recoverable Resource Manager Services Attach Facility (RRSAF).

A unit of recovery is indoubt when a failure occurs after a successful prepare but before a successful commit. This number should equal the number of units of recovery gone indoubt. If it is less, then some indoubt units of recovery might still exist.

Field Name: Q3STRIUR

COMMIT PH 1
The number of successful requests for commit phase 1 in a two-phase commit environment such as CICS or IMS. It includes successfully prepared agents associated with threads that use the Recoverable Resource Manager Services Attach Facility (RRSAF). It does not include successful single-phase commits or distributed two-phase commits.

Background and Tuning Information
IMS and CICS applications use the PREPARE and COMMIT sequence to commit work.

Field Name: Q3STPREP

SSAM EOM
The number of times MVS deleted non-DB2 address space while connected to DB2.
IFCID 001 - Subsystem Services Data

Field Name: Q3STMEOM
TERMIN.THREAD
The number of time threads that terminated successfully.
This number does not agree with the create thread count because each level of a thread's access (IDENTIFY, SIGNON, and CREATE THREAD) must be terminated.

Background and Tuning Information
The value of this field is usually greater than the number of create thread occurrences, because it also includes the termination of connections to DB2 (IDENTIFY) and other internal counts.

Field Name: Q3STTERM
EXITS
The number of successful DSN3EXIT requests.

Field Name: Q3STEXIT
SYNCHS
The total number of commits in a single-phase commit environment such as TSO, CAF, or UTILITY. IMS applications use the prepare-and-commit sequence; CICS applications use both the synchronized commit request and the prepare-and-commit sequence to commit work.
Note that DBATs executed on this location are not included. For DBAT statistics, see SINGLE PHASE COMMITS received on the DDF activity block.

Field Name: Q3STSYNC
SSAM EOT
The number of times non-DB2 tasks abended while connected to DB2.

Field Name: Q3STMEOT
CRT.THRD QUED
The number of create thread requests queued (not including DBATs).
The total number of threads accessing data that can be allocated concurrently is the MAX USERS value on the installation panel DSNTIPE. Requests are queued when the MAX USERS value is exceeded. If no threads are queued during peak hours, the maximum number of threads might be set too high.

Background and Tuning Information
As a rule of thumb about 1% thread queuing is acceptable. When this is appreciably higher, increase the value of MAX USERS on the DB2 install panel DSNTIPE.
The combined maximum allowed for MAX USERS and MAX REMOTE ACTIVE cannot exceed 2000.

Field Name: Q3STCTHW
This is an exception field.

SUBS.INT.CALLS
The number of subsystem interface calls processed.
Field Name: Q3STSSSI

READ ONLY COMMIT

The number of read-only commits.

There are occasions when CICS or IMS invokes DB2 when no DB2 resource was altered since the completion of the last commit process. When this occurs, DB2 performs both phases of the two-phase commit during the first commit phase and records that the user or job is read-only in relation to its DB2 processing.

Field Name: Q3STRDON

IDBACK*

The maximum number of connections to a single instance from batch or TSO background tasks.

This is a high-water mark.

Field Name: Q3STHWIB

IDFORE*

The maximum number of connections to a single instance from TSO foreground tasks.

This is a high water-mark.

Field Name: Q3THWIF

CTREAD*

The highest number of batch CICS, IMS, and TSO tasks (CTREAD) to a single instance.

This is a high-water mark.

Field Name: Q3THWCT
IFCID 001 - QSST Data

This topic shows detailed information about “Record Trace - IFCID 001 - QSST Data”.

This block contains DB2 serviceability fields. Most of these fields are for IBM service use.

Record trace - IFCID 001 - QSST Data

The field labels shown in the following sample layout of “Record Trace - IFCID 001 - QSST Data” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>QSSTGPLF</td>
<td>97</td>
</tr>
<tr>
<td>QSSTFPLF</td>
<td>10</td>
</tr>
<tr>
<td>QSSTFREF</td>
<td>4</td>
</tr>
<tr>
<td>QSSTEXPF</td>
<td>1143</td>
</tr>
<tr>
<td>QSSTCONF</td>
<td>2</td>
</tr>
<tr>
<td>QSSTGPLV</td>
<td>227</td>
</tr>
<tr>
<td>QSSTFPLV</td>
<td>24</td>
</tr>
<tr>
<td>QSSTFREV</td>
<td>13</td>
</tr>
<tr>
<td>QSSTEXPV</td>
<td>159</td>
</tr>
<tr>
<td>QSSTCONFV</td>
<td>6</td>
</tr>
<tr>
<td>QSSTGETM</td>
<td>708</td>
</tr>
<tr>
<td>QSSTFREEM</td>
<td>607</td>
</tr>
<tr>
<td>QSSTCONTZ</td>
<td>0</td>
</tr>
<tr>
<td>QSSTCONT</td>
<td>0</td>
</tr>
<tr>
<td>QSSTCRIT</td>
<td>0</td>
</tr>
<tr>
<td>QSSTABND</td>
<td>0</td>
</tr>
<tr>
<td>QSSTSGETM</td>
<td>435</td>
</tr>
<tr>
<td>QSSTSGETR</td>
<td>91244</td>
</tr>
<tr>
<td>QSSTFREEM</td>
<td>4</td>
</tr>
<tr>
<td>QSSTFREER</td>
<td>90807</td>
</tr>
<tr>
<td>QSSTD4POST</td>
<td>0</td>
</tr>
<tr>
<td>QSSTM4DISNUM</td>
<td>0</td>
</tr>
<tr>
<td>QSSTM4DISPGS</td>
<td>0</td>
</tr>
<tr>
<td>QSSTFREER64</td>
<td>0</td>
</tr>
<tr>
<td>QSSTFREER64</td>
<td>0</td>
</tr>
<tr>
<td>QSSTD4DISNUM</td>
<td>0</td>
</tr>
<tr>
<td>QSSTD4DISPGS</td>
<td>0</td>
</tr>
<tr>
<td>QSSTD4DISNUM</td>
<td>0</td>
</tr>
<tr>
<td>QSSTD4DISPGS</td>
<td>0</td>
</tr>
<tr>
<td>QSSTRSMAXWARN</td>
<td>0</td>
</tr>
<tr>
<td>QSSTP64DISNM</td>
<td>39</td>
</tr>
<tr>
<td>QSSTP64DISPGS</td>
<td>39</td>
</tr>
<tr>
<td>QSSTP64DISNM</td>
<td>1985</td>
</tr>
<tr>
<td>QSSTP64DISPGS</td>
<td>39</td>
</tr>
<tr>
<td>QSSTCONTSTORNUM</td>
<td>0</td>
</tr>
</tbody>
</table>

**QSSTGPLF**

This field is for IBM service use.

**Field Name:** QSSTGPLF

**QSSTCONT**

The number of full system contractions.

**Field Name:** QSSTCONT

**QSSTCRIT**

The number of critical storage shortages after contraction.

**Field Name:** QSSTCRIT

**QSSTABND**

The number of abends due to local storage shortage.

**Field Name:** QSSTABND
IFCID 001 - QVAS Data

IFCID 001 - QVAS Data

This topic shows detailed information about “Record Trace - IFCID 001 - QVAS Data”.

This block contains DB2 serviceability fields. These fields are for IBM service use.

Record trace - IFCID 001 - QVAS Data

The field labels shown in the following sample layout of “Record Trace - IFCID 001 - QVAS Data” are described in the following section.

QVAS DATA

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>QVASSUSP</td>
<td>32742</td>
</tr>
<tr>
<td>QVASSUSR</td>
<td>622</td>
</tr>
<tr>
<td>QVASSUS</td>
<td>348</td>
</tr>
<tr>
<td>QVASSUT</td>
<td>19</td>
</tr>
<tr>
<td>QVASSUSM</td>
<td>180</td>
</tr>
<tr>
<td>QVASSAUT</td>
<td>67</td>
</tr>
<tr>
<td>QVASCBO</td>
<td>99999</td>
</tr>
<tr>
<td>QVASCBOF</td>
<td>99999</td>
</tr>
<tr>
<td>QVASCBO</td>
<td>99999</td>
</tr>
<tr>
<td>QVASCBOF</td>
<td>99999</td>
</tr>
</tbody>
</table>

QVASSUSP

This field is for IBM service use.

Field Name: QVASSUSP
IFCID 001 - QVLS Data

This topic shows detailed information about “Record Trace - IFCID 001 - QVLS Data”.

The QVLS latch counters represent the number of suspends that were performed by agents that attempted to obtain a latch.

There is not a one-to-one relationship between the QVLS counters and IFCID 56 or 57, because an agent might suspend multiple times or not at all, while trying to obtain a latch. That is why the QVLS counters are not directly related to Accounting Class 3.

Record trace - IFCID 001 - QVLS Data

The field labels shown in the following sample layout of “Record Trace - IFCID 001 - QVLS Data” are described in the following section.

QVLS DATA

<table>
<thead>
<tr>
<th>QVLSLC01</th>
<th>0</th>
<th>QVLSLC02</th>
<th>0</th>
<th>QVLSLC03</th>
<th>0</th>
<th>QVLSLC04</th>
<th>0</th>
<th>QVLSLC05</th>
<th>0</th>
<th>QVLSLC06</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>QVLSLC07</td>
<td>0</td>
<td>QVLSLC08</td>
<td>0</td>
<td>QVLSLC09</td>
<td>0</td>
<td>QVLSLC10</td>
<td>0</td>
<td>QVLSLC11</td>
<td>0</td>
<td>QVLSLC12</td>
<td>0</td>
</tr>
<tr>
<td>QVLSLC13</td>
<td>0</td>
<td>QVLSLC14</td>
<td>0</td>
<td>QVLSLC15</td>
<td>0</td>
<td>QVLSLC16</td>
<td>0</td>
<td>QVLSLC17</td>
<td>0</td>
<td>QVLSLC18</td>
<td>0</td>
</tr>
<tr>
<td>QVLSLC19</td>
<td>3</td>
<td>QVLSLC20</td>
<td>0</td>
<td>QVLSLC21</td>
<td>0</td>
<td>QVLSLC22</td>
<td>0</td>
<td>QVLSLC23</td>
<td>1907</td>
<td>QVLSLC24</td>
<td>131</td>
</tr>
<tr>
<td>QVLSLC25</td>
<td>0</td>
<td>QVLSLC26</td>
<td>0</td>
<td>QVLSLC27</td>
<td>19</td>
<td>QVLSLC28</td>
<td>0</td>
<td>QVLSLC29</td>
<td>3</td>
<td>QVLSLC30</td>
<td>25</td>
</tr>
<tr>
<td>QVLSLC31</td>
<td>4</td>
<td>QVLSLC32</td>
<td>147</td>
<td>QVLSLC33</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**QVLSLC01**

This field is infrequently used.

**Field Name:** QVLSLC01

**QVLSLC02**

The predominant latch usage is: Global authorization cache.

**Field Name:** QVLSLC02

**QVLSLC03**

The predominant latch usage is: DDF disconnect.

**Field Name:** QVLSLC03

**QVLSLC04**

This field is infrequently used.

**Field Name:** QVLSLC04

**QVLSLC05**

The predominant latch usage is: IRLM data sharing exits or RLF.

**Field Name:** QVLSLC05

**QVLSLC06**

The predominant latch usage is: Data sharing index split.

**Field Name:** QVLSLC06

**QVLSLC07**

The predominant latch usage is: Index lotch and OBD allocation.

**Field Name:** QVLSLC07

**QVLSLC08**

The predominant latch usage is: Query parallelism.
IFCID 001 - QVLS Data

Field Name: QVLSLC08
QVLSLC09
The predominant latch usage is: Utilities or stored procedure URIDs.
Field Name: QVLSLC09
QVLSLC10
The predominant latch usage is: Allied agent chain or sequence descriptors.
Field Name: QVLSLC10
QVLSLC11
This field is infrequently used.
Field Name: QVLSLC11
QVLSLC12
The predominant latch usage is: Global transaction ID table.
Field Name: QVLSLC12
QVLSLC13
The predominant latch usage is: Pageset operations.
Field Name: QVLSLC13
QVLSLC14
The predominant latch usage is: Bufferpool LRU.
Field Name: QVLSLC14
QVLSLC15
The predominant latch usage is: ARCHIVE LOG MODE(QUIESCE).
Field Name: QVLSLC15
QVLSLC16
This field is infrequently used.
Field Name: QVLSLC16
QVLSLC17
The predominant latch usage is: RURE chain.
Field Name: QVLSLC17
QVLSLC18
The predominant latch usage is: DDF resynch list.
Field Name: QVLSLC18
QVLSLC19
The predominant latch usage is: Log write.
Field Name: QVLSLC19
QVLSLC20
The predominant latch usage is: System checkpoint.
Field Name: QVLSLC20
QVLSLC21
The predominant latch usage is: Accounting rollup.
Field Name: QVLSLC21

QVLSLC22
The predominant latch usage is: Internal checkpoint.
Field Name: QVLSLC22

QVLSLC23
The predominant latch usage is: Buffer manager.
Field Name: QVLSLC23

QVLSLC24
The predominant latch usage is: EDM pool or prefetch.
Field Name: QVLSLC24

QVLSLC25
The predominant latch usage is: Workfile allocation.
Field Name: QVLSLC25

QVLSLC26
The predominant latch usage is: Dynamic statement cache.
Field Name: QVLSLC26

QVLSLC27
The predominant latch usage is: Stored procedures or authorization cache.
Field Name: QVLSLC27

QVLSLC28
The predominant latch usage is: Stored procedures or authorization cache.
Field Name: QVLSLC28

QVLSLC29
The predominant latch usage is: Field procs and DDF transaction manager.
Field Name: QVLSLC29

QVLSLC30
The predominant latch usage is: Agent services.
Field Name: QVLSLC30

QVLSLC31
The predominant latch usage is: Storage manager.
Field Name: QVLSLC31

QVLSLC32
The predominant latch usage is: Storage manager.
Field Name: QVLSLC32

QVLSLC254
The predominant latch usage is: Index latch.

Field Name: QVLSLC254
IFCID 001 - z/OS Metrics

This topic shows detailed information about “Record Trace - IFCID 001 - z/OS Metrics”.

Record trace - IFCID 001 - z/OS Metrics

The field labels shown in the following sample layout of “Record Trace - IFCID 001 - z/OS Metrics” are described in the following section.

Z/OS METRICS

LPAR CPS

The number of central processors (CPs) on the logical partition (LPAR).

Field Name: QWOSLNCP

LPAR CPU UTILIZATION

The CPU utilization on the LPAR.

Field Name: QWOSLPRU

DB2 SUBSYS CPU UTILIZATION

The CPU utilization of the DB2 subsystem.

Field Name: QWOSDB2U

LPAR PAGE-IN RATE

The LPAR page-in rate.

Field Name: QWOSLPIR

DB2 SUBSYS PAGE-IN RATE

The DB2 subsystem page-in rate.

Field Name: QWOSDPIR

LPAR REAL STOR (MB)

The amount of real storage available on the LPAR.

Field Name: QWOSLRST

DB2 SUBSYS USED REAL STOR (MB)

The amount of real storage used for the DB2 subsystem.

Field Name: QWOSDRSU

LPAR VIRT STOR (MB)

The amount of virtual storage available on the LPAR.

Field Name: QWOSLVST

DB2 SUBSYS USED VIRT STOR (MB)
IFCID 001 - z/OS Metrics

The amount of virtual storage used for the DB2 subsystem.
Field Name: QWOSDVSU

LPAR FREE REAL STOR (MB)
The amount of free real storage on the LPAR.
Field Name: QWOSLRSF

DB2 MSTR CPU UTILIZATION
The CPU utilization of the DB2 MSTR address space.
Field Name: QWOSMSTU

LPAR FREE VIRT STOR (MB)
The amount of free virtual storage on the LPAR.
Field Name: QWOSLVSF

DB2 DBM1 CPU UTILIZATION
The CPU utilization of the DB2 DBM1 address space.
Field Name: QWOSDBMU

UNREFERENCED INTERVALS
The number of unreferenced intervals.
Field Name: QWOSLUIC

QWOSFLG
This field is for IBM service use.
Field Name: QWOSFLG

QWOSRCDE
This field is for IBM service use.
Field Name: QWOSRCDE

QWOSRSNC
This field is for IBM service use.
Field Name: QWOSRSNC
IFCID 002 - DB2 Statistics

Database 2 Statistics shows the data from IFCID 002.

“IFCID 002 - Accelerator Data” on page 40-64
This topic shows detailed information about “Record Trace - IFCID 002 - Accelerator Data”.

“IFCID 002 - Buffer Pool Activity” on page 40-69
This topic shows detailed information about “Record Trace - IFCID 002 - Buffer Pool Activity”.

“IFCID 002 - Data Manager Data” on page 40-85
This topic shows detailed information about “Record Trace - IFCID 002 - Data Manager Data”.

“IFCID 002 - Data Sharing Locking Data” on page 40-92
This topic shows detailed information about “Record Trace - IFCID 002 - Data Sharing Locking Data”.

“IFCID 002 - Dynamic SQL Statement” on page 40-95
This topic shows detailed information about “Record Trace - IFCID 002 - Dynamic SQL Statement”.

“IFCID 002 - EDM Pool Data” on page 40-97
This topic shows detailed information about “Record Trace - IFCID 002 - EDM Pool Data”.

“IFCID 002 - Group Buffer Pools Activity Data” on page 40-103
This topic shows detailed information about “Record Trace - IFCID 002 - Group Buffer Pools Activity Data”.

“IFCID 002 - Locking Data” on page 40-112
This topic shows detailed information about “Record Trace - IFCID 002 - Locking Data”.

“IFCID 002 - Miscellaneous” on page 40-116
This topic shows detailed information about “Record Trace - IFCID 002 - Miscellaneous”.

“IFCID 002 - Nested SQL Activity” on page 40-118
This topic shows detailed information about “Record Trace - IFCID 002 - Nested SQL Activity”.

“IFCID 002 - Query Parallelism” on page 40-120
This topic shows detailed information about “Record Trace - IFCID 002 - Query Parallelism”.

“IFCID 002 - RID List Processing” on page 40-124
This topic shows detailed information about “Record Trace - IFCID 002 - RID List Processing”.

“IFCID 002 - ROWID” on page 40-126
This topic shows detailed information about “Record Trace - IFCID 002 - ROWID”.

“IFCID 002 - Service Controller Data” on page 40-127
This topic shows detailed information about “Record Trace - IFCID 002 - Service Controller Data”.

“IFCID 002 - SQL Call Data” on page 40-135
This topic shows detailed information about “Record Trace - IFCID 002 - SQL Call Data”.

Chapter 40. IFCID Record Blocks
**IFCID 002 - Accelerator Data**

**IFCID 002 - Accelerator Data**

This topic shows detailed information about “Record Trace - IFCID 002 - Accelerator Data”.

**Record trace - IFCID 002 - Accelerator Data**

The field labels shown in the following sample layout of “Record Trace - IFCID 002 - Accelerator Data” are described in the following section.

### ACCELERATOR DATA

**PRODUCT ID:** AQT02012  
**SERVER ID:** EMU03  
**STATE:** ONLINE

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>QUERIES SUCCESSFULLY EXECUTED</td>
<td>205</td>
</tr>
<tr>
<td>QUERIES FAILED TO EXECUTE</td>
<td>0</td>
</tr>
<tr>
<td>CURRENTLY EXECUTING QUERIES</td>
<td>0</td>
</tr>
<tr>
<td>MAXIMUM EXECUTING QUERIES</td>
<td>2</td>
</tr>
<tr>
<td>CONNECTS TO ACCELERATOR</td>
<td>29</td>
</tr>
<tr>
<td>REQUESTS SENT TO ACCELERATOR</td>
<td>667</td>
</tr>
<tr>
<td>BLOCS SENT TO ACCELERATOR</td>
<td>0</td>
</tr>
<tr>
<td>BLOCKS RECEIVED FROM ACCELERATOR</td>
<td>609</td>
</tr>
<tr>
<td>TCP/IP SERVICES ELAPSED TIME</td>
<td>1:15:15.660640</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRODUCT ID</td>
<td>AQT02012</td>
</tr>
<tr>
<td>SERVER ID</td>
<td>EMU03</td>
</tr>
<tr>
<td>STATE</td>
<td>ONLINE</td>
</tr>
</tbody>
</table>

### PRODUCT ID

The accelerator product identifier.  
**Field Name:** Q8STPRID

### SERVER ID

The accelerator server identifier.  
**Field Name:** Q8STNAME

### STATE

The current accelerator state:

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>INITIALIZED</td>
</tr>
<tr>
<td>1</td>
<td>ONLINE</td>
</tr>
<tr>
<td>2</td>
<td>PAUSED</td>
</tr>
<tr>
<td>3</td>
<td>OFFLINE</td>
</tr>
<tr>
<td>4</td>
<td>STOPPED</td>
</tr>
<tr>
<td>5</td>
<td>MAINTENANCE</td>
</tr>
<tr>
<td>6</td>
<td>DOWN</td>
</tr>
<tr>
<td>7</td>
<td>UNKNOWN</td>
</tr>
</tbody>
</table>

**Field Name:** Q8STTATE

### QUERIES SUCCESSFULLY EXECUTED

The number of queries (sent by this DB2 system since accelerator start) that were successfully executed in the accelerator.
Field Name: Q8STSREQ

QUERIES FAILED TO EXECUTE
The number of queries (sent by this DB2 system since accelerator start) that failed to be successfully executed for any reason, including the accelerator being in an invalid state.

Field Name: Q8STFREQ

QUERIES FAILED TO EXECUTE - ACCELERATOR IN INVALID STATE
The number of queries (sent by this DB2 system since accelerator start) that failed to be successfully executed, for example, because the accelerator was in an invalid state.

Field Name: Q8STFINV

CURRENTLY EXECUTING QUERIES
The number of currently executing queries in the accelerator. This includes the queries from all the DB2 systems connected to this accelerator.

Field Name: Q8STACTV

MAXIMUM EXECUTING QUERIES
The maximum number of queries executing in the accelerator at any time since accelerator start. This includes the queries from all the DB2 systems connected to this accelerator.

Field Name: Q8STMAXA

CONNECTS TO ACCELERATOR
The number of connects to the accelerator from this DB2 system.

Field Name: Q8STCONN

REQUESTS SENT TO ACCELERATOR
The number of Distributed Relational Database Architecture™ (DRDA) requests sent by this DB2 system to the accelerator.

Field Name: Q8STREQ

REQUESTS SENT TO ACCELERATOR - TIMED OUT
The number of connections that were timed out when this DB2 system sent requests to the accelerator.

Field Name: Q8STTOUT

REQUESTS SENT TO ACCELERATOR - FAILED
The number of connections that failed when this DB2 system sent requests to the accelerator.

Field Name: Q8STFAIL

BYTES SENT TO ACCELERATOR
The total number of bytes sent to the accelerator.

Field Name: Q8STBYTS

BYTES RECEIVED FROM ACCELERATOR
The total number of bytes received from the accelerator.

Field Name: Q8STBYTR
MESSAGES SENT TO ACCELERATOR
The total number of messages sent to the accelerator.
Field Name: Q8STMSGS

MESSAGES RECEIVED FROM ACCELERATOR
The total number of messages received from the accelerator.
Field Name: Q8STMSGR

BLOCKS SENT TO ACCELERATOR
The total number of blocks sent to the accelerator.
Field Name: Q8STBLKS

BLOCKS RECEIVED FROM ACCELERATOR
The total number of blocks received from the accelerator.
Field Name: Q8STBLKR

ROWS SENT TO ACCELERATOR
The total number of rows sent to the accelerator.
Field Name: Q8STROWS

ROWS RECEIVED FROM ACCELERATOR
The total number of rows received from the accelerator.
Field Name: Q8STROWR

TCP/IP SERVICES ELAPSED TIME
The accumulated accelerator services TCP/IP elapsed time measured in DB2. It starts when sending the requests to the accelerator and ends when receiving the results from the accelerator.
Field Name: Q8STTELA

WAIT TIME IN ACCELERATOR
The accumulated wait time spent in the accelerator when executing requests from the DB2 subsystem.
Field Name: Q8STAWAT

AVG. QUEUE LENGTH (LAST 3 HOURS)
The average query queue length during the last 3 hours at the accelerator.
Field Name: Q8STAVGQ03

AVG. QUEUE LENGTH (LAST 24 HOURS)
The average query queue length during the last 24 hours at the accelerator.
Field Name: Q8STAVGQ24

MAXIMUM QUEUE LENGTH
The high watermark of the query queue length at the accelerator.
Field Name: Q8STMAXQ

AVG QUEUE WAIT ELAPSED TIME
The average wait time at the accelerator query queue.
ICCID 002 - Accelerator Data

Field Name: Q8STQUEW
MAX QUEUE WAIT ELAPSED TIME
The maximum wait time at the accelerator query queue.
Field Name: Q8STQUEM

WORKER NODES
The number of active worker nodes.
Field Name: Q8STWNOD

WORKER NODES AVG CPU UTILIZATION (%) 
The average CPU utilization on accelerator worker nodes.
Field Name: Q8STWCPU

COORDINATOR AVG CPU UTILIZATION (%) 
The average CPU utilization on the accelerator coordinator node.
Field Name: Q8STCCPU

DISK STORAGE AVAILABLE (MB) 
The disk storage (MB) available at the accelerator.
Field Name: Q8STDSKA

DISK STORAGE AVAILABLE - IN USE (%) 
The disk storage in-use at the accelerator, expressed as a percentage (%).
Field Name: Q8STDSKU

DISK STORAGE AVAILABLE - IN USE FOR DATABASE (MB) 
The disk storage in-use for the DB2 subsystem.
Field Name: Q8STDSKB

DATA SLICES 
The number of data slices at the accelerator.
Field Name: Q8STNMDS

DATA SKEW 
When table data is loaded into the accelerator, it may be unevenly distributed across the different data slices on the disks. This disparity is called data skew. The counter represents the accumulated skew over all tables that belong to the DB2 subsystem. The skew of a table is the ratio that shows how uneven the data slices are, as calculated by ((maximum data slice size - minimum data slice size) / median data slice size).
A high value indicates that data reorganization can improve disk utilization and query performance.
Field Name: Q8STSK EW

PROCESSORS 
The total amount of processors within the accelerator.
Field Name: Q8STCORS

ELAPSED TIME IN ACCELERATOR
ICCID 002 - Accelerator Data

The accumulated elapsed time spent in the accelerator when executing requests from the DB2 subsystem.

Field Name: Q8STAE LA

CPU TIME SPENT IN ACCELERATOR

The accumulated CPU time spent in the accelerator when executing requests from the DB2 subsystem.

Field Name: Q8STACPU
IFCID 002 - Buffer Pool Activity

This topic shows detailed information about “Record Trace - IFCID 002 - Buffer Pool Activity”.

This block shows buffer pool activity at thread level.

For details on buffer pool activities, refer to the documentation of Performance Expert Buffer Pool Analyzer.

This report has the same layout as "IFCID 002 - Miscellaneous” on page 40-116.

Record trace - IFCID 002 - Buffer Pool Activity

The field labels shown in the following sample layout of “Record Trace - IFCID 002 - Buffer Pool Activity” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>BUFFER POOL ID</td>
<td></td>
</tr>
<tr>
<td>CURRENT ACTIVE BUFFERS</td>
<td></td>
</tr>
<tr>
<td>BUFFER UPDATES</td>
<td></td>
</tr>
<tr>
<td>GETPAGE REQUESTS-SEQUENTIAL</td>
<td></td>
</tr>
<tr>
<td>NUMBER OF DATA SET OPENS</td>
<td></td>
</tr>
<tr>
<td>SYNCHRONOUS WRITES</td>
<td></td>
</tr>
<tr>
<td>SYNCHRONOUS READS-SEQUENTIAL</td>
<td></td>
</tr>
<tr>
<td>DFNH MIGRATED DATA SETS</td>
<td></td>
</tr>
<tr>
<td>VPOOL EXPANSION OR CONTRACT</td>
<td></td>
</tr>
<tr>
<td>LIST PREFETCH REQUESTS</td>
<td></td>
</tr>
<tr>
<td>LIST PREFETCH READS</td>
<td></td>
</tr>
<tr>
<td>PREFETCH I/O STREAMS REDUCTION</td>
<td></td>
</tr>
<tr>
<td>MAX WORKFILES CONCURRENTLY USED</td>
<td></td>
</tr>
<tr>
<td>DYNAMIC PREFETCH REQUESTS</td>
<td></td>
</tr>
<tr>
<td>DYNAMIC PREFETCH READS</td>
<td></td>
</tr>
<tr>
<td>VPOOL EXPANSION OR CONTRACT</td>
<td></td>
</tr>
<tr>
<td>WORKFILE NOT WRITTEN</td>
<td></td>
</tr>
<tr>
<td>MINIMUM BUFFERS ON SLRU (LWP)</td>
<td></td>
</tr>
<tr>
<td>MAXIMUM BUFFERS ON SLRU (HAM)</td>
<td></td>
</tr>
<tr>
<td>RANDOM GETPAGE BUFFER HIT</td>
<td></td>
</tr>
</tbody>
</table>

BUFFER POOL ID

The buffer pool ID.

Field Name: QBSTPID

FLAGS

The flag byte shows if more QBST data is following or if this is the last of the QBST repeating groups.

Field Name: QBSTFLG

CURRENT ACTIVE BUFFERS

The total number of currently active (nonstealable) buffers. This field is an instantaneous sample of the number of buffers in the buffer pool that were updated or in use at the time this monitor data was requested. Because this field gives a snapshot value at statistics collection time, it only shows a problem if it happens at this time.

Background and Tuning Information

The buffer pool might be too small if the percentage of active pages in the buffer pool is beyond the deferred write threshold (DWQT).

Field Name: QBSTCBA
GETPAGE REQUESTS

The number of Getpage requests including conditional and unconditional requests.

Field Name: QBSTGET

BUFFER UPDATES

The number of times buffer updates were requested against pages in the buffer pool.

Background and Tuning Information

The ratio of Buffer Updates to Pages Written (QBSTPWS) suggests a high level of efficiency as it increases, because more updates are being externalized per physical write.

Buffer updates per pages written depends strongly on the type of application. For example, a batch program that processes a table in skip sequential mode with a high row update frequency in a dedicated environment can achieve very good update efficiency. In contrast, update efficiency tends to be lower for transaction processing applications, because transaction processing tends to be random.

The following can influence the number of updates per page:

Number of rows per page

A small PCTFREE value gathers more rows on the same page. However, at the same time this can impact concurrency.

Buffer pool size and deferred write thresholds

Increase DWQT and VDWQT or the size of the buffer pool. This causes DB2 to let page updates accumulate in the buffer pool. Therefore, the probability that more updates per page get captured increases. This effect is less significant if the buffer pool is concurrently used by several transactions, it also depends on the type of transaction.

Field Name: QBSTSWS

This is an exception field.

UNAVAILABLE BUFFER-VPOOL FULL

The number of times a usable buffer could not be located in the virtual buffer pool because the virtual buffer pool was full.

Background and Tuning Information

Ideally, this value should be 0. Any other value indicates that the buffer pool is underallocated. In this case, use the ALTER BUFFERPOOL command to increase the virtual buffer pool size until this value remains at 0.

Field Name: QBSTXFL

This is an exception field.

GETPAGE REQUESTS-SEQUENTIAL

The number of Getpage requests issued by sequential access requesters.

Field Name: QBSTSGT

PAGES WRITTEN
Background and Tuning Information

Consider the ratio of Pages Written per write I/O. The number of write I/O operations includes Asynchronous Writes (QBSTWIO) and Synchronous Writes (QBSTIMW).

The ratio of pages per write I/O suggests a high level of efficiency as the ratio increases, because more pages are being externalized per physical write.

The following factors impact the ratio of pages written per write I/O:

Checkpoint frequency
At checkpoint time, I/Os are scheduled to write all updated pages on the deferred write queue to DASD. If this occurs too frequently, the deferred write queue does not grow large enough to achieve a high ratio of pages written per write I/O.

The checkpoint frequency depends on the number of logs written between two consecutive checkpoints. This number is set at installation time; see the field CHECKPOINT FREQ of installation panel DSNTIPN.

Frequency of active log switch
DB2 takes a system checkpoint each time the active log is switched. High frequency of active log switches causes the problem described under checkpoint frequency.

Buffer pool size and deferred write thresholds
The deferred write thresholds (VDWQT and DWQT) are a function of buffer pool size. If the buffer pool size is decreased, these thresholds are reached more frequently, causing I/Os to be scheduled more often to write some of the pages on the deferred write queue to DASD. This prevents the deferred write queue from growing large enough to achieve a high ratio of pages written per write I/O.

Number of data sets, and the spread of updated pages across them
The efficiency of write I/O also depends on the number of data sets associated with the buffer pool and spread of updated pages across them. Because of the nature of batch processing, the ratio of pages written to write I/Os can be expected to be higher than that expected for transaction type workloads.

To determine update efficiency, use also the value in the Buffer Updates field (QBSTSW) to check the number of buffer updates per page written.

Field Name: QBSTPWS
This is an exception field.

NUMBER OF DATA SET OPENS
The number of data sets physically opened successfully. This value is cumulative from the start of the DB2 statistics interval.

Field Name: QBSTDSO
This is an exception field.

SYNCHRONOUS READS
IFCID 002 - Buffer Pool Activity

The number of synchronous read I/O operations performed by DB2 for applications and utilities.

Background and Tuning Information
This number includes both Synchronous Reads Sequential Access Only (QBSTSIQ) and synchronous read operations for non-sequential access.

You can use this value and the value of Synchronous Reads Sequential Access Only to calculate the number of Non-Sequential Synchronous Reads.

Check the buffer pool hit ratio if the number of non-sequential synchronous reads is larger than expected.

Field Name: QBSTRIQ
This is an exception field.

SYNCHRONOUS WRITES
The total number of immediate writes.
Immediate writes occur when:
• An immediate write threshold is reached
• No deferred write engines are available
• More than two checkpoints pass without a page being written.

Sometimes DB2 uses synchronous writes even when the IWTH is not exceeded. As an example, when more than two checkpoints pass without a page being written. This type of situation does not indicate a buffer shortage.

Background and Tuning Information
A small number of immediate writes can be expected. Synchronous writes occur if there are too many checkpoints and/or the buffer pool is too small.

If a large number of synchronous writes occur, monitor the DM Critical Threshold Reached (QBSTDQMC) field. Reaching Immediate Write Threshold (IWTH-97.5%) implies that the Data Management Threshold (DMTH-95%) has been crossed. You can ignore the value in the immediate write field when DM Critical Threshold Reached is zero. Otherwise consider increasing the size of the buffer pool. You can use the ALTER BUFFERPOOL command. However, the original buffer pool attributes reappear when DB2 stops and restarts.

Check also the System Event Checkpoint field (QWSDCKPT) in the Subsystem Services block to see whether the frequency of DB2 checkpoints should be reduced. To do this, increase the value of ZPARM LOGLOAD.

Field Name: QBSTIMW
This is an exception field.

BUFFERS ALLOCATED-VPOOL
The number of buffers allocated for a virtual buffer pool.
The number of buffers within each pool is always less than or equal to the corresponding value specified at installation time or when using the ALTER BUFFERPOOL command.

Background and Tuning Information
You should monitor the buffer pool hit ratio field to find the optimum size of the buffer pool. Usually the buffer pool hit ratio is improved by increasing the size of the buffer pool. However, paging the buffer pool storage impacts DB2 performance if the virtual buffer pool is too large.

Page-ins Required for Read I/O (QBSTRPI) and Page-ins Required for Write I/O (QBSTWPI) are useful when determining whether paging affects the performance of a certain buffer pool. The Resource Measurement Facility (RMF) also provides reports on MVS paging activity:

**Storage Paging**
When the virtual buffer pool is extended into expanded storage, MVS storage paging activity occurs. If a large buffer pool size results in excessive storage paging, consider using hiperpools.

**Paging to Auxiliary Storage**
If the virtual buffer pool size requirements exceed the central storage and expanded storage available, the oldest buffer pool pages migrate to auxiliary paging storage. When these pages are accessed subsequently, I/O must bring them back into real storage. This should be avoided. You could have a smaller buffer pool and let DB2 do the I/O rather than use MVS paging with its I/O CPU overhead. This is a situation that you (as the system programmer) should monitor.

You can use the ALTER BUFFERPOOL command to alter the size of the virtual buffer pool. However, the original buffer pool attributes reappear when DB2 stops and restarts.

Changing the size of the virtual buffer pool implicitly changes the buffer pool thresholds. See the Deferred Write Threshold Reached field (QBSTDWT).

**Field Name:** QBSTVPL

**SYNCHRONOUS READS-SEQUENTIAL**

The number of synchronous read I/O requests issued by sequential access requesters.

**Background and Tuning Information**

Sequential synchronous read I/Os can occur because:

- Prefetch is disabled (QBSTSPD).
- Prefetch pages could have been stolen from the buffer pool before the Getpage request is issued for those pages. Subsequently the pages are reread synchronously. A negative buffer pool hit ratio can indicate the same problem.
- The pages requested are not consecutive: DB2 estimated the selected range of pages to be so small that prefetch would make no sense. See also Sequential Prefetch Requested (QBSTSEQ).

It is normal to have a small value for SYNC READ I/O (SEQUENTIAL) because before the sequential prefetch is scheduled, the first page of a prefetch is read by SYNC READ I/O. However, if this number is large, consider increasing the size of the buffer pool or reviewing the sequential steal thresholds (VPSEQT and HPSEQT).

**Field Name:** QBSTSIO

This is an *exception* field.
IFCID 002 - Buffer Pool Activity

ASYNCHRONOUS WRITES
The number of asynchronous write I/O operations performed by media manager to a direct access storage device.

Field Name: QBSTWIO
This is an exception field.

DFHSM MIGRATED DATA SETS
The number of times migrated data sets were encountered.

Field Name: QBSTMIG
This is an exception field.

SEQUENTIAL PREFETCH REQUESTS
The number of sequential prefetch requests. This counter is incremented for each PREFETCH request (which can result in an I/O read). If the prefetch results in an I/O read, up to 32 pages may be read for SQL, and up to 64 pages for utilities. A request does not result in an I/O read if all pages to be prefetched are already in the buffer pool.

This counter does not include sequential detection, which is recorded in the Dynamic Prefetch - Requested field.

Background and Tuning Information
Sequential prefetch reads a sequential set of pages. It allows CP and I/O operations to be overlapped. DB2 determines at BIND time whether sequential prefetch is used or not.

Sequential prefetch is generally used for a table space scan. It can also be used to read index pages in an index scan. For an index scan that accesses 8 or more consecutive data pages, DB2 requests sequential prefetch at bind time. The index must have a cluster ratio of 80% or higher. (Use REORG and RUNSTATS and rebind relevant SQL if you doubt that this target has been met previously.)

The number of prefetch requests by itself is not a good indicator for efficiency of prefetching:

- At run time not every prefetch request results in read I/O: the Sequential Prefetch Reads field (QBSTPIO) shows the number of read I/O operations caused by sequential prefetch. The Prefetch Disabled No Buffer (QBSTSPD) and Prefetch Disabled No Read Engine fields (QBSTREE) show the number of times prefetch was disabled because buffers and read engines had not been available.

- Check the value in the buffer pool hit ratio. A negative value indicates that prefetched pages are stolen from the buffer pool before they are read. The pages are subsequently reread synchronously. There will be also a large value in the Synchronous Reads Total (QBSTRIO) field.

- Decreasing the size of the buffer pool can reduce the prefetch quantity, leading to a larger number of prefetch requests. See also the Sequential Prefetch Pages Read field (QBSTTPP).

Field Name: QBSTSEQ
This is an exception field.

HORIZONTAL DEFERRED WRITE THRESHOLD
The number of times the deferred write threshold (DWTH) was reached.
This threshold is a percentage of the virtual buffer pool that might be occupied by unavailable pages, including both updated pages and pages in use. DB2 checks this threshold when an update to a page is completed. If the percentage of unavailable pages in the virtual buffer pool exceeds the threshold, write operations are scheduled for enough data sets (up to 128 pages per data set) to reduce the number of unavailable buffers to 10% below the threshold.

Background and Tuning Information

The default value for this threshold is 50%. You can change that to any value from 0% to 90% by using the DWQT option on the ALTER BUFFERPOOL command.

The deferred write thresholds, DWQT and VDWQT, are specified as a percentage, their absolute value depends on the size of the virtual buffer pool.

Consider the following aspects when changing the deferred write thresholds:

**Optimize the ratio of pages written per write I/O**

The ratio can be monitored using the Pages Written (QBSTPWS) field.

When the buffer pool is relatively small, the default thresholds could prevent the deferred write queue from growing large enough to achieve a high ratio of pages written per write I/O. Raising these thresholds will, in this instance, reduce the I/O write frequency, increasing the number of pages written per I/O.

**Distribute I/O evenly over time**

If a virtual buffer pool is very large, it is unlikely that the default values of either DWQT or VDWQT will ever be reached. In this case, write I/Os tend to occur in surges, triggered by DB2 checkpoints. Lowering the VDWQT and the DWQT could improve performance by distributing the write I/Os more evenly over time.

**Impact on other buffer pool thresholds**

Increasing DWQT and VDWQT allows updated pages to use a larger portion of the virtual buffer pool. Large DWQT and VDWQT can have a significant effect on the other thresholds. For example, in work load where pages are frequently updated, and the updated pages exceed the size of the virtual buffer pool, setting both DWQT and VDWQT to 90% would probably cause frequent threshold-reached events for sequential prefetch (and possibly the data management and immediate write).

Field Name: QBSTDW

This is an exception field.

**DFHSM RECALL TIMEOUTS**

The number of recall timeouts.

Field Name: QBSTRTO

**SEQUENTIAL PREFETCH READS**

The number of asynchronous read I/O operations due to normal sequential prefetch (applications and utilities).
Prefetch Read I/O is not activated if one of the following conditions applies:

- All pages in the prefetch range are already in the buffer pool.
- Prefetch is disabled (QBSTSPD).

This means that the value in this field is usually smaller than the number of sequential prefetch requests (QBSTSEQ).

**Field Name:** QBSTPIO

This is an exception field.

**VERTICAL DEFERRED WRITE THRESHOLD**

The number of times the vertical deferred write threshold (VDWQT) was reached. This threshold is similar to the deferred write threshold but it applies to the number of updated pages for one single page set in the buffer pool. If the percentage or number of updated pages for the data set exceeds the threshold, writes up to 128 pages are scheduled for that data set.

**Field Name:** QBSTDWV

This is an exception field.

**VPOOL EXPANSION OR CONTRACT**

The number of successful virtual buffer pool expansions or contractions due to the ALTER BUFFERPOOL command. An increase in this counter indicates that buffer-pool-related system parameters have been changed.

**Field Name:** QBSTVPA

This is an exception field.

**PAGES READ VIA SEQUENTIAL PREFETCH**

The total number of pages read due to a normal sequential prefetch. A sequential prefetch request does not result in a read I/O if all the desired pages are found in the buffer pool.

**Background and Tuning Information**

The ratio of Sequential Prefetch Pages Read to Sequential Prefetch Reads (QBSTPIO) is usually between 0 and 32.

For requests issued by application programs, the number of pages per READ I/O primarily depends on the page size and the size of the buffer pool. Normally thirty-two 4 KB pages (or four 32 KB pages) is the maximum prefetch quantity for table space scans, whether data or index. Utilities use a prefetch quantity of up to 64 pages.

The number of pages per READ I/O can be lower because:

- Pages within the prefetch range may already be in the buffer pool.
- Not enough pages are available because of a buffer shortage.
- A prefetch quantity of 8 pages or less is used for work files.

A small value for this ratio can indicate:

- A good performing buffer pool being so large that most of the pages, which had otherwise to be prefetched, are cached in the buffer pool. In this case, the buffer pool hit ratio should be high.
- A buffer shortage condition, reducing the efficiency of sequential prefetch. This could mean, for example, work-file prefetch quantity...
reduction from 8 to 4 to 2, as the number of available buffers shrinks. In this case, you should consider tuning the buffer pool.

Field Name: QBSTSPP
This is an exception field.

DATA MANAGER BUF CRITICAL THRESHOLD
The number of times the data manager critical threshold (DMTH-95%) was reached.
This field shows how many times a page was immediately released because the data management threshold was reached.
The threshold is checked before a page is read or updated. If the threshold has not been exceeded, DB2 accesses the page in the virtual buffer pool once for each page, no matter how many rows are retrieved or updated in that page. If the threshold has been exceeded, Getpage requests and RELEASEs apply to rows instead of pages. That is, if more than one row is retrieved or updated in a page, more than one Getpage request and RELEASE is performed on that page.

Background and Tuning Information
Avoid reaching this threshold wherever possible because it significantly affects CPU usage. Set virtual buffer pool sizes large enough or reduce the workload on the buffer pool.

Field Name: QBSTDRC
This is an exception field.

VPOOL OR HPOOL EXPANSION FAILURE
The total number of virtual buffer pool expansion failures due to the lack of virtual storage space.

Background and Tuning Information
Ideally, this value should be 0. If it is not, check the virtual storage allocation of the DB2 database address space for areas that can be reduced. For example, you can reduce the size of other buffer pools.

Field Name: QBSTXFV
This is an exception field.

LIST PREFETCH REQUESTS
The number of list prefetch requests.
List prefetch allows DB2 to access data pages efficiently even when the required data pages are not contiguous. It allows CP and I/O operations to be overlapped.

Background and Tuning Information
DB2 determines at BIND time whether sequential prefetch is used. List prefetch is chosen as follows:
• Usually with a single index that has a cluster ratio lower than 80%.
• Sometimes on a single index with a high cluster ratio, if the estimated amount of data to be accessed is too small to make sequential prefetch efficient.
• Always to access data by multiple index access.
IFCID 002 - Buffer Pool Activity

- Always to access data from the inner table during a hybrid join.

DB2 never chooses list prefetch if the estimated number of RIDs to be processed takes more than 50% of the RID pool. During execution time, list prefetch processing terminates if more than 25% of the rows (with a minimum of 4075) in the table must be accessed.

Data pages are read in quantities equal to the sequential prefetch quantity (QBSTSEQ), which depends on buffer pool size and is usually 32 pages.

**Field Name:** QBSTLPF

This is an *exception* field.

**CONCURRENT PREFETCH I/O STREAMS-HWM**

The highest number of concurrent prefetch I/O streams allocated to support a parallel I/O or CP query in this buffer pool. It reflects prefetch activities for non-workfile page sets.

This number only applies to query I/O and CP parallelism.

**Field Name:** QBSTXIS

This is an *exception* field.

**LIST PREFETCH READS**

The number of asynchronous read I/O operations caused by the list prefetch.

The number of pages read is shown by the List Prefetch Pages Read (QBSTLPP) field.

**Background and Tuning Information**

Prefetch Read I/O is not activated if one of the following conditions apply:

- All pages in the prefetch range are already in the buffer pool.
- Prefetch is disabled (Prefetch Disabled No Read Engine - QBSTREE).

This means that the value in this field is usually less than the number of list prefetch requests (QBSTLPF).

**Field Name:** QBSTLIO

This is an *exception* field.

**PAGE-INS REQUIRED FOR WRITE**

The number of page-ins required for a write I/O.

**Field Name:** QBSTWPI

**PREFETCH I/O STREAMS REDUCTION**

The total number of requested prefetch I/O streams that were denied because of a lack of buffer pool storage space.

It only applies to query I/O and CP parallelism.

For example, if 100 prefetch I/O streams are requested and only 80 are granted, then 20 is added to the number in this field.

**Background and Tuning Information**

Consider increasing the size of the buffer pool if this value is not 0.

The ratio of this field and the Reduced parallel query requests field gives the average degree of parallel query processing that was reduced because
IFCID 002 - Buffer Pool Activity

of insufficient buffer pool space. The Prefetch I/O streams - Concurrent streams - high-water mark field gives the highest degree of parallel query processing that was reduced for one or more queries processed in parallel.

The number in this field reflects the prefetch activities for non-workfile page sets.

Field Name: QBSTJIS
This is an exception field.

PAGES READ VIA LIST PREFETCH
The number of pages read via list prefetch.

Field Name: QBSTLPP

MAX WORKFILES CONCURRENTLY USED
The maximum number of work files concurrently used during merge processing within this statistics period.

Ideally, each work file needs 16 buffers to allow DB2 to perform a sequential prefetch for work files.

Field Name: QBSTWFM
This is an exception field.

PARALLEL QUERY REQUESTS
The total number of requests made for parallel query support in this buffer pool. This field only applies to non-workfile page sets in query I/O and CP parallelism.

Field Name: QBSTPQO

DYNAMIC PREFETCH REQUESTS
The number of dynamic prefetch requests. Dynamic prefetch is the process that is triggered because of sequential detection. If the prefetch request results in an I/O read, up to 32 advancing pages can be read at a time.

Background and Tuning Information
Dynamic prefetch reads a sequential set of pages. It allows CP and I/O operations to be overlapped. If DB2 does not choose prefetch at bind time it can sometimes use it at execution time. The method is called sequential detection.

The number of prefetch requests by itself is not a good indicator for efficiency of prefetching because:

- At run time not every prefetch request results in read I/O: the Dynamic Prefetch Reads field shows the number of read I/O operations caused by dynamic prefetch. The Prefetch Disabled No Buffer (QBSTSPD) and Prefetch Disabled No Read Engine (QBSTREE) fields show the number of times prefetch was disabled because buffers and read engines had not been available.

- Prefetch pages can be stolen from the buffer pool before they are read. This is indicated by a negative buffer pool hit ratio. The pages are subsequently reread synchronously. This will also cause an unexpectedly large value for total synchronous reads (QBSTRIO).

Decreasing the size of the buffer pool can reduce the prefetch quantity (QBSTDPP), leading to a larger number of prefetch requests.
IFCID 002 - Buffer Pool Activity

Field Name: QBSTDPF
This is an exception field.

MERGE PASSES REQUESTED
The total number of merge passes for DB2 sort activities. This value reflects how many merge passes were requested for DB2 to determine the number of work files permitted to support each merge pass.

Field Name: QBSTWFR

DYNAMIC PREFETCH READS
The number of asynchronous read I/Os because of dynamic prefetch. The number of pages read is recorded in the Dynamic Prefetch Pages Read field.

Background and Tuning Information
A prefetch request does not result in an I/O if one of the following conditions apply:

- All pages to be prefetched are already in the buffer pool.
- The prefetch is canceled.

This means that the value in this field is usually smaller than the number of dynamic prefetch requests.

Field Name: QBSTDIO
This is an exception field.

MERGE PASS DEGRADED-LOW BUFFER
The number of times that a merge pass was not efficiently performed due to a shortage of space in the buffer pool. The number in this field is incremented for each merge pass where the maximum number of work files allowed is less than the number of work-files requested.

Background and Tuning Information
The maximum number of work files allowed is calculated as follows:

- Buffers consumed = 2 * (work files already allocated)
- Buffers available = (sequential steal threshold * buffer pool size - buffers consumed)
- Maximum work files allowed = buffers available / (2 * 8)

The default for the sequential steal threshold is 0.8.

Ideally, the number in this field should be 0. Otherwise, it indicates a shortage of buffer pool space or that there are too many concurrent work files. For example, there could be a number of concurrently open cursors that require sorting. Consider increasing the buffer pool size using the ALTER BUFFERPOOL command.

Field Name: QBSTWFF
This is an exception field.

PAGES READ VIA DYNAMIC PREFETCH
The number of pages read because of dynamic prefetch. Dynamic prefetch is the process that is triggered by sequential detection.

Background and Tuning Information
The ratio of Dynamic Prefetch Pages Read to Dynamic Prefetch Reads is between 0 and 32.

DB2 can fetch up to 32 pages per prefetch.

The number of pages per READ I/O can be lower because:
- Pages within the prefetch range are already in the buffer pool.
- Not as many pages are available due to a buffer shortage.

A small value for this ratio can indicate:
- A good performing buffer pool being large enough to contain pages that would otherwise be prefetched. This is indicated by a high buffer pool hit ratio.
- A buffer shortage condition, which reduces the efficiency of dynamic prefetch. In this instance the buffer pool hit ratio will be low. Consider tuning the buffer pool.

**Field Name:** QBSTDPP

This is an *exception* field.

**WORKFILE REQUEST REJECTED-LOW BUFFER**

The total number of work files that were rejected during all merge passes because of insufficient buffer resources.

**Background and Tuning Information**

This field and the degraded low buffers field determine the average number of work files that cannot be honored at each merge pass because of insufficient buffer pool space.

Ideally, the number in this field should be 0. Otherwise, it indicates a shortage of buffer pool space or that there are too many concurrent work files. For example, there could be a number of concurrently open cursors that require sorting. Consider increasing the size of the buffer pool using the ALTER BUFFERPOOL command.

Note that, when there are many concurrent sorts or large sorts, it is a good idea to dedicate a separate buffer pool for sort work files. This will greatly facilitate work-file performance tuning.

**Field Name:** QBSTWFD

This is an *exception* field.

**WORKFILE REQUESTED-ALL MERGE PASS**

The total number of work files requested for all merge passes.

This field and the Merge Passes Requested field determine the average number of work files requested in a single merge pass.

For DB2 to perform an efficient prefetch for work files, each workfile should have at least 16 dedicated buffers. Work files used during sort phase processing or other non-sort-related processing are not included in this number.

**Field Name:** QBSTWFT

**WORKFILE NOT CREATED-NO BUFFER**

This field is only applicable if DB2 is running under MVS/XA.
The number of times a work file could not be created due to insufficient buffer resources. It indicates that a sort is in progress and limited in regard to the number of work files it can use.

**Background and Tuning Information**

Ideally, this should be 0. Otherwise, it indicates a shortage of buffer pool space or that there are many concurrent work files. For example, there could be a number of open cursors that require sorting.

Generally, sorts are performed more efficiently with additional work files, but there are internal DB2 limits on the number of work files a transaction can have. It is possible that at run time a transaction cannot use as many work files as it had planned. You can control this by increasing the buffer pool size (ALTER BUFFERPOOL), or changing the transaction so it requires fewer concurrent work files.

**Field Name:** QBSTMAX

This is an *exception* field.

**PREFETCH QUANTITY REDUCED TO HALF**

The total number of times prefetch quantity is reduced from normal to 50% of normal. The normal size depends on the page size of the buffer pool.

This field only applies to query I/O and CP parallelism.

**Background and Tuning Information**

The number in this field indicates when DB2 had to reduce the sequential prefetch quantity to continue executing concurrently with parallel queries in the system. If the number is small, it may be tolerable.

**Field Name:** QBSTPL1

This is an *exception* field.

**PREFETCH DISABLED-NO BUFFER**

The total number of times sequential prefetch was disabled because buffers were not available.

**Field Name:** QBSTSPD

This is an *exception* field.

**WORKFILE PREFETCH NOT SCHEDULED**

The number of times a sequential prefetch was not scheduled for a work file because the dynamic prefetch quantity is zero.

**Background and Tuning Information**

The work-file prefetch checks the dynamic prefetch quantity (normally 1 to 8 pages). When the quantity is zero, the value in this field is incremented. A high number in this field implies that the buffer pool is too small.

Ideally, the number in this field should be 0. Otherwise, it indicates a shortage of buffer pool space or that there are many concurrent work files. For example, there could be a number of concurrently open cursors that require sorting.

Consider increasing the size of the buffer pool or allocating a buffer pool specifically for DSNDB07 usage. This can be especially effective with high-use query systems whose reports make extensive use of sort activity.
Field Name: QBSTWKPD
This is an exception field.

**PREFETCH QUANTITY REDUCED TO QUARTER**

The total number of times prefetch quantity is reduced from 50% to 25% of normal. The normal size depends on the page size of the buffer pool.

This field only applies to query I/O and CP parallelism.

**Background and Tuning Information**

The query response for parallel queries can be significantly degraded if the value in this field is not 0.

Field Name: QBSTPL2
This is an exception field.

**PREFETCH DISABLED-NO READ ENGINE**

The total number of times a prefetch is disabled because of an unavailable read engine.

**Background and Tuning Information**

Because there are 600 read engines, a maximum of 600 concurrent prefetch operations can be processed at a time. When this maximum is reached, prefetching is disabled and this count is incremented. The value in this field should be close to 0.

Field Name: QBSTREE
This is an exception field.

**WORKFILE PAGES TO DESTRUCT**

The number of pages for which destructive read was requested.

Field Name: QBSTWDRP

**FAILED COND SEQ&RDM GETPAGE REQUEST**

The number of sequential and random Getpage requests which failed because the page was not in the buffer pool. Failed conditional requests do not initiate I/O operations.

Field Name: QBSTNGT

**PAGE-INS REQUIRED FOR READ**

The number of page-ins required for a read I/O.

Field Name: QBSTRPI

**WORKFILE PAGES NOT WRITTEN**

The number of pages dequeued from VDWQ for destructive read requests.

Field Name: QBSTWBVQ

**FAILED COND SEQ GETPAGE REQUEST**

The number of conditional sequential Getpage requests which failed because the page was not in the buffer pool. Failed conditional requests do not initiate I/O operations.

Field Name: QBSTNSG

**MINIMUM BUFFERS ON SLRU (LWM)**
The minimum number of buffers on the sequential least-recently-used (SLRU) chain in the last statistical period. This is the low-water mark (LWM) within an interval.

Field Name: QBSTSMIN

PAGES ADDED TO LPL

The number of times that one or more pages were added to the logical page list (LPL).

Field Name: QBSTLPL

MAXIMUM BUFFERS ON SLRU (HWM)

The maximum number of buffers on the sequential least-recently-used (SLRU) chain in the last statistical period. This is the high-water mark (HWM) within an interval.

Field Name: QBSTSMAX

LENGTH OF SLRU = VPSEQT

The number of times when the length of the sequential least-recently-used (SLRU) chain equals the sequential steal threshold VPSEQT.

Field Name: QBSTHST

RANDOM GETPAGE BUFFER HIT

The number of times that the random Getpage request has a buffer hit and the buffer is on the least-recently-used (SLRU) chain.

Field Name: QBSTRHS
### Record trace - IFCID 002 - Data Manager Data

The field labels shown in the following sample layout of “Record Trace - IFCID 002 - Data Manager Data” are described in the following section.

**DATA MANAGER DATA**

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CUR RIDLIST BLOCKS</td>
<td>The number of RID blocks currently in use (snapshot value).</td>
</tr>
<tr>
<td>MAX RIDLIST BLOCKS</td>
<td>The highest number of RID blocks in use at any time since DB2 startup. This is a high-water mark.</td>
</tr>
<tr>
<td>RIDLIST TERMINATED-RDS LIMIT</td>
<td>The number of times RID list processing terminated because the number of RIDs that can fit into the guaranteed number of RID blocks was greater than the maximum limit (25% of table size).</td>
</tr>
</tbody>
</table>

#### CUR RIDLIST BLOCKS

The number of RID blocks currently in use (snapshot value).

**Field Name:** QISTRCUR

#### CUR RIDLIST BLOCKS OVERFLOWED

The number of RID blocks currently residing in work file storage. This is a snapshot value.

**Field Name:** QISTWFRCUR

#### MAX RIDLIST BLOCKS

The highest number of RID blocks in use at any time since DB2 startup. This is a high-water mark.

**Field Name:** QISTRHIG

#### MAX RIDLIST BLOCKS OVERFLOWED

The maximum number of RID blocks overflowed (stored) to a work file at any time since DB2 start. This is a high-water mark.

**Field Name:** QISTWFRHIG
**Background and Tuning Information**

Ideally, this value should be 0.

The matching index scan part of the RID list processing scanned more than 25% of the index. RID list processing is then terminated, the index scan is abandoned and normally replaced by a tablespace scan.

Reasons for this are:

- Inaccurate or incomplete RUNSTATS statistics. To avoid this, you should collect all statistics on a regular basis, especially simple and correlated column statistics. Using RUNSTATS with SHRLEVEL(CHANGE) does not prevent access to data.
- Optimizer error. In this instance, you could disable RID list processing by adding the clause OPTIMIZE FOR 1 ROW to the SQL statement, or force the access path to index only by adding the necessary columns to the index.

Field Name: QISTRLLM

This is an exception field.

**RIDLIST TERMINATED-DM LIMIT**

The number of times a RID list processing operation terminated because the number of RID entries was greater than the physical limit of approximately 26 million RIDs.

Field Name: QISTRPLM

This is an exception field.

**RIDLIST TERMINATED-NO STORAGE**

The number of times RID list processing was not used because DBM1 storage was exhausted.

**Background and Tuning Information**

This failure occurs when the DBM1 storage limit is reached.

Field Name: QISTRSTG

This is an exception field.

**RIDLIST TERMINATED-PROC.LIMIT**

The number of times the maximum RID pool storage was exceeded.

The size is determined by the installation parameter RID POOL SIZE (DB2 install panel DSNTIPC). It can be 0, or between 128 KB and 10 GB. The general formula for calculating the RID pool size is:

\( (\text{Number of concurrent RID processing activities}) \times (\text{average number of RIDs}) \times 2 \times (5 \text{ bytes per RID}) \).

Field Name: QISTRMAX

This is an exception field.

**COLUMNS BYPASSED**

The total number of columns (rows x columns) for which an invalid select procedure was encountered.

DB2 bypasses invalid select procedures which can cause some degradation in performance.
Field Name: QISTCOLS
This is an exception field.

TOTAL WHOLE STORAGE (MB) - (DB2 9)
Total whole megabytes of storage that are currently used in the Workfile database.

Field Name: QISTWFCU

TOTAL FRACT STORAGE (KB) - (DB2 9)
Total fractions of megabytes of storage, specified in kilobytes, that are currently used in the workfile database.

Field Name: QISTWFCK

MAX TOTAL STORAGE (MB) - (DB2 9)
Maximum total amount of storage that is used in the Workfile database.

Field Name: QISTWFMU

AGENT MAX STORAGE (MB) - (DB2 9)
Maximum total amount of storage that is used in the Workfile database.

Field Name: QISTWFMU

TOTAL WHOLE 4K STORAGE (MB) - (DB2 9)
The number of times the maximum amount of storage that an agent can use in the Workfile database was exceeded.

Field Name: QISTWFNE

TOTAL WHOLE 32K STORAGE (MB) - (DB2 9)
Total whole megabytes of storage that were used for 32 KB table spaces.

Field Name: QISTWF32

TOTAL FRACT 4K STORAGE (KB) - (DB2 9)
The total whole megabytes of storage that were used for 4 KB table spaces.

Field Name: QISTWF04K

TOTAL FRACT 32K STORAGE (KB) - (DB2 9)
Total whole megabytes of storage that were used for 32 KB table spaces.

Field Name: QISTWF32K

CUR TOTAL STORAGE USED (KB)
The total amount of storage (KB) currently used in the Workfile Database at system level.

Field Name: QISTWCTO

MAX AGENT STORAGE LIMIT (KB)
The maximum amount of storage (KB) in the Workfile Database that can be used by each agent (derived from ZPARM MAXTEMPS).

Field Name: QISTWMXA

MAX TOTAL STORAGE USED (KB)
**IFCID 002 - Data Manager Data**

- The maximum total amount of storage (KB) ever used in the Workfile Database at system level since DB2 startup.
  - Field Name: QISTWMXU

**MAX STORAGE USAGE LIMIT EXCEEDED**

- The number of times the maximum amount of storage that an agent can use in the Workfile database was exceeded.
  - Field Name: QISTWFNE

**CUR 4K TABSPACE STORAGE USED (KB)**

- The total amount of storage (KB) currently used for 4 KB table spaces in the Workfile Database.
  - Field Name: QISTW4K

**CUR 32K TABSPACE STORAGE USED (KB)**

- The total amount of storage (KB) currently used for 32 KB table spaces in the Workfile Database.
  - Field Name: QISTW32K

**4K INSTEAD OF 32K TABSPACE USED**

- The number of times that space in a 4 KB page table space was used because space in a 32 KB page table space was preferred but not available in the Workfile Database.
  - Field Name: QISTWFP2

**32K INSTEAD OF 4K TABSPACE USED**

- The number of times that space in a 32 KB page table space was used because space in a 4 KB page table space was preferred but not available in the Workfile Database.
  - Field Name: QISTWFP1

**CUR ACTIVE (DM) IN-MEMORY**

- The number of currently active in-memory work files created by the Data Manager.
  - Field Name: QISTIMAC

**CUR STORAGE (DM) IN-MEMORY (KB)**

- The total space used for currently active in-memory work files created by the Data Manager.
  - Field Name: QISTIMSC

**MAX ACTIVE (DM) IN-MEMORY**

- The maximum number of in-memory work files (created by the Data Manager) that were active at any point in time since DB2 startup. This is a high-water mark count.
  - Field Name: QISTIMAH

**MAX STORAGE (DM) IN-MEMORY (KB)**

- The maximum space used for active in-memory work files created by the Data Manager at any point in time since DB2 startup. This is a high-water mark count.
Field Name: QISTIMSH
CUR ACTIVE (SORT) IN-MEMORY
The number of currently active in-memory work files created by the SORT component.

Field Name: QISTSIAC
CUR STORAGE (SORT) IN-MEMORY (KB)
The total space used for currently active in-memory work files created by the SORT component.

Field Name: QISTSISC
MAX ACTIVE (SORT) IN-MEMORY
The maximum number of in-memory work files created by the SORT component that were active at any point in time since DB2 start. This is a high-water mark count.

Field Name: QISTSIAH
MAX STORAGE (SORT) IN-MEMORY (KB)
The maximum space used for active in-memory work files created by the SORT component at any point in time since DB2 startup. This is a high-water mark count.

Field Name: QISTDGTTCTO
CUR ACTIVE (NONSORT) IN-MEMORY
The number of currently active non-SORT related in-memory work files created by the Data Manager.

Field Name: QISTI2AC
CUR DGTT STORAGE USED (KB)
The total amount of storage (KB) currently used for DGTTs in the Workfile Database by all agents on the system.

Field Name: QISTI2AH
MAX ACTIVE (NONSORT) IN-MEMORY
The maximum number of non-SORT related in-memory work files created by the Data Manager that were active at any point in time since DB2 startup. This is a high-water mark count.

Field Name: QISTDGTTMXU
IN-MEMORY (NONSORT) OVERFLOWED
The number of times non-SORT related in-memory work files overflowed into a physical table space.

Field Name: QISTI2OF
CUR WORKFILE STORAGE USED (KB)
IFCID 002 - Data Manager Data

The total amount of storage (KB) currently used for non-DGTT work files in the Workfile Database by all agents on the system.

Field Name: QISTWFCTO

IN-MEMORY WORKFILE NOT CREATED

The number of times an in-memory work file was not created due to critical storage conditions.

Field Name: QISTIMNC

MAX WORKFILE STORAGE USED

The maximum total amount of storage (KB) ever used for non-DGTT work files in the Workfile Database by all agents on the system since DB2 startup.

Field Name: QISTWFMXU

TOTAL STORAGE CONFIG (KB)

The total storage (KB) configured for all table spaces in the Workfile Database.

Field Name: QISTWSTG

MAX AGENT STORAGE USED (KB)

The maximum amount of storage (KB) ever used in the Workfile Database by any thread since DB2 startup.

Field Name: QISTAMXU

TOTAL DGTT STORAGE CONFIG (KB)

The total preferred storage (KB) configured for DGTTs in the Workfile Database.

Field Name: QISTDGTTSTG

AGENT STORAGE THRESHOLD (%)

The alert threshold of high space-usage for DGTTs or non-DGTT work files in the Workfile Database by an agent (derived from ZPARM WFSTGUSE_AGENT_THRESHOLD).

Field Name: QISTASTH

TOTAL WORKFILE STORAGE CONFIG (KB)

The total preferred storage (KB) configured for non-DGTT work files in the Workfile Database.

Field Name: QISTWFSTG

TOTAL STORAGE THRESHOLD (%)

The alert threshold of high space-usage for DGTTs or non-DGTT work files in the Workfile Database (derived from zparm WFSTGUSE_SYSTEM_THRESHOLD).

Field Name: QISTSTSSTH

INSERT ROWS SKIPPED

The number of rows skipped by read transactions because of uncommitted INSERT operations (using currently committed semantic for FETCH).

Field Name: QISTRCCI
IFCID 002 - Data Manager Data

DELETE ROWS ACCESSED

The number of rows accessed by read transactions because of uncommitted DELETE operations (using currently committed semantic for FETCH).

Field Name: QISTRCCD

UPDATE ROWS ACCESSED

The number of rows accessed by read transactions because of uncommitted UPDATE operations (using currently committed semantic for FETCH).

Field Name: QISTRCCU
IFCID 002 - Data Sharing Locking Data

This topic shows detailed information about “Record Trace - IFCID 002 - Data Sharing Locking Data”.

Record trace - IFCID 002 - Data Sharing Locking Data

The field labels shown in the following sample layout of “Record Trace - IFCID 002 - Data Sharing Locking Data” are described in the following section.

LOCK REQ (P-LOCKS)

The number of lock requests for P-locks.

Field Name: QTGSPLPLK

SYNCH.XES - LOCK REQ

The number of P/L-lock requests propagated to z/OS XES synchronously.
This number is not incremented if the request is suspended before going to XES.

Field Name: QTGSLSLM

SUSPENDS - IRLM GLBL CONT

The number of suspensions due to IRLM global resource contention. All IRLM lock states were in conflict on the same resource.

Global contention requires intersystem communication to resolve the lock conflict whereas local contention does not.

Field Name: QTGSIGLO

UNLOCK REQ (P-LOCKS)

The number of unlock requests for P-locks.

Field Name: QTGSUPLK

SYNCH.XES - CHANGE REQ

The number of change requests propagated to z/OS XES synchronously, including logical and physical locks.
This number is not incremented if the request is suspended before going to XES.

Field Name: QTGSCSLM

SUSPENDS - XES GLBL CONT

The number of suspensions due to z/OS XES global resource contention. The z/OS XES lock states were in conflict but the IRLM lock states were not.

IRLM has many lock states but XES is only aware of the exclusive and shared lock states.

Field Name: QTGSSGLO
CHANGE REQ (P-LOCKS)

The number of change requests for P-locks.

Field Name: QTGSCPLK

SYNCH.XES - UNLOCK REQ

The number of unlock requests propagated to z/OS XES synchronously, including logical and physical locks.

This number is not incremented if the request is suspended before going to XES.

Field Name: QTGSUSLM

NOTIFY MESSAGES SENT

The number of notify messages sent.

Field Name: QTGSNTFY

ASYNCH.XES - RESOURCES

The number of resources propagated by IRLM to z/OS XES asynchronously, including logical and physical locks.

This can happen when new inter-DB2 interest occurs on a parent resource or when a request completes after the requester's execution unit was suspended.

Field Name: QTGSKIDS

INCOMPAT RETAINED LOCK

The number of global lock or change requests denied or suspended due to an incompatible retained lock.

Field Name: QTGSDRTA

NOTIFY MESSAGES RECEIVED

The number of notify messages received.

Field Name: QTGSNTFR

P-LOCK/NFY EX.ENGINES

The maximum number of engines available for physical lock exit or notify exit requests.

Field Name: QTGSPEMX

P-LOCK/NFY EX.ENGINE N/A

The number of times an engine is not available for physical lock exit or notify exit requests.

Field Name: QTGSPEQW

PSET/PART P-LOCK NEGOTIAT.

The number of times this DB2 was driven to negotiate a partition or page set physical lock due to changing inter-DB2 interest levels on the partition or page set.

Field Name: QTGSPPPE

PAGE P-LOCK NEGOTIAT.
IFCID 002 - Data Sharing Locking Data

The number of times this DB2 negotiated a page physical lock because of physical lock contention within DB2.

**Field Name:** QTGSPGPE

**OTHER P-LOCK NEGOTIATION**

The number of times this DB2 was driven to negotiate a physical lock type other than page set, partition, or page.

**Field Name:** QTGSOTPE

**P-LOCK CHANGE DURING NEG.**

The number of times a physical lock change request was issued during physical lock negotiation.

**Field Name:** QTGSCHNP

**FALSE CONTENTIONS**

The total number of false contentions for LOCK and UNLOCK requests. A false contention occurs when different resource names hash to the same entry in the coupling facility (CF) lock table. The CF detects contention within the hash entry, and XES uses intersystem messaging to determine that no actual resource contention exists.

**Note:** The QTGSFCON flag indicates whether the false contention is reported at subsystem (=1) or LPAR level (=0).

**Field Name:** QTGSFLMG

**SYNC-ASYNC XES CONV**

The number of synchronous to asynchronous heuristic conversions for LOCK requests in XES. This conversion is done when XES determines that it is more efficient to drive the request asynchronously to the coupling facility (CF).

**Field Name:** QTGSFLSE

**FLMG COUNTS PER**

Flags describing QTGS counters:

- **ON**  QTGSFLMG counts per subsystem (SUBSYS)
- **OFF**  QTGSFLMG counts per LPAR

**Field Name:** QTGSFLGS

**NO DELAY LOCK REQ REJECTS**

The total number of failed DB2 lock requests to XES to process without delay. XES rejects the lock request because it could not process it synchronously.

**Field Name:** QTGSCREJ
Record trace - IFCID 002 - Dynamic SQL Statement

The field labels shown in the following sample layout of “Record Trace - IFCID 002 - Dynamic SQL Statement” are described in the following section.

<table>
<thead>
<tr>
<th>Field Label</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>DYNAMIC SQL STMT</td>
<td></td>
</tr>
<tr>
<td>REOPTIMIZATION</td>
<td>0 FOUND IN CACHE 10 IMPLICIT PREPARES 0</td>
</tr>
<tr>
<td>SYMT INVALID (MAX)</td>
<td>0 NOT FOUND IN CACHE 0 PREPARES AVOIDED 0</td>
</tr>
<tr>
<td>CSWL STMTS PARSED</td>
<td>0 CSWL LITS REPLACED 0 CSWL MATCHES FOUND 0</td>
</tr>
<tr>
<td>CSWL DUPLS CREATED</td>
<td></td>
</tr>
</tbody>
</table>

REOPTIMIZATION

The total number of times reoptimization occurs because the value of the host variable or parameter marker changes.

Field Name: QXSTREOP

FOUND IN CACHE

The number of times a PREPARE command was satisfied by copying a statement from the prepared statement cache.

Field Name: QXSTFND

IMPLICIT PREPARES

An implicit prepare occurs when the user copy of the prepared SQL statement no longer exists in the local dynamic SQL cache and the application plan or package is bound with KEEPDYNAMIC YES.

If the skeleton copy of the prepared SQL statement exists in the global dynamic SQL cache in the EDM pool, a short prepare is executed, otherwise a full prepare is executed.

Field Name: QXSTIPRP

STMT INVALID (MAX)

The number of times statements are invalidated in the local dynamic SQL cache because the MAXKEEPD limit has been reached and prepared SQL statements in the local dynamic SQL cache have to be reclaimed.

Field Name: QXSTDEXP

NOT FOUND IN CACHE

The number of times that DB2 searched the prepared statement cache but could not find a suitable prepared statement.

Field Name: QXSTNFND

PREPARES AVOIDED

This field indicates the number of times where no SQL PREPARE or EXECUTE IMMEDIATE was issued by the application and a copy of a prepared SQL statement was found in local dynamic SQL cache.

When an application plan or package is bound with KEEPDYNAMIC YES, a copy of each prepared SQL statement for the application thread is held in the local dynamic SQL cache and kept across a commit boundary.
An application thread can save the total cost of a prepare by using a copy of the prepared statement in the local dynamic SQL cache from an earlier prepare by the same thread. To do this, the application must be modified to avoid issuing repetitive SQL PREPAREs for the same SQL statement.

**Field Name:** QXSTNPRP

**STMT INVALID (DDL)**
The number of times statements are invalidated in the local dynamic SQL cache because of SQL DDL or updated RUNSTATS information and prepared SQL statements in the local dynamic SQL cache have to be reclaimed.

**Field Name:** QXSTDINV

**CSWL STMTS PARSED**
The number of times DB2 parsed dynamic statements because CONCENTRATE STATEMENTS WITH LITERALS behavior was used for the prepare of the statement for the dynamic statement cache.

**Field Name:** QXSTCWLP

**CSWL LITS REPLACED**
The number of times DB2 replaced at least one literal in a dynamic statement because CONCENTRATE STATEMENTS WITH LITERALS was used for the prepare of the statement for dynamic statement cache.

**Field Name:** QXSTCWLR

**CSWL MATCHES FOUND**
The number of times DB2 found a matching reusable copy of a dynamic statement in cache during prepare of a statement that had literals replaced because of CONCENTRATE STATEMENTS WITH LITERALS.

**Field Name:** QXSTCWLM

**CSWL DUPLS CREATED**
The number of times DB2 created a duplicate STMT instance in the statement cache for a dynamic statement that had literals replaced by CONCENTRATE STATEMENTS WITH LITERALS behavior. The duplicate STMT instance was needed because a cache match failed because the literal reusability criteria was not met.

**Field Name:** QXSTCWLD
### Record trace - IFCID 002 - EDM Pool Data

The field labels shown in the following sample layout of “Record Trace - IFCID 002 - EDM Pool Data” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>EDM POOL DATA</td>
<td></td>
</tr>
<tr>
<td>PAGES IN POOL :</td>
<td>0</td>
</tr>
<tr>
<td>CT PAGES :</td>
<td>0</td>
</tr>
<tr>
<td>PT PAGES :</td>
<td>0</td>
</tr>
<tr>
<td>FREE PAGES :</td>
<td>0</td>
</tr>
<tr>
<td>CT REQUESTS :</td>
<td>0</td>
</tr>
<tr>
<td>PT REQUESTS :</td>
<td>31</td>
</tr>
<tr>
<td>DBD REQUESTS :</td>
<td>231</td>
</tr>
<tr>
<td>EDM POOL FULL :</td>
<td>0</td>
</tr>
<tr>
<td>CT NOT IN POOL :</td>
<td>4</td>
</tr>
<tr>
<td>PT NOT IN POOL :</td>
<td>19</td>
</tr>
<tr>
<td>DBD NOT IN POOL :</td>
<td>3</td>
</tr>
<tr>
<td>CACHE INSERTS :</td>
<td>8</td>
</tr>
<tr>
<td>CACHE REQUESTS :</td>
<td>24</td>
</tr>
<tr>
<td>PKG SEARCH NOT FOUND INSERT:</td>
<td>3</td>
</tr>
<tr>
<td>PKG SEARCH NOT FOUND DELETE:</td>
<td>0</td>
</tr>
<tr>
<td>Statements in Global Cache :</td>
<td>0</td>
</tr>
<tr>
<td>PAGES IN STMT POOL (ABOVE) :</td>
<td>28346</td>
</tr>
<tr>
<td>HELD BY STATEMENTS :</td>
<td>25</td>
</tr>
<tr>
<td>FREE PAGES :</td>
<td>28321</td>
</tr>
<tr>
<td>FAILS DUE TO STMT POOL FULL :</td>
<td>0</td>
</tr>
<tr>
<td>PAGES IN RDS POOL (ABOVE) :</td>
<td>0</td>
</tr>
<tr>
<td>HELD BY CT :</td>
<td>0</td>
</tr>
<tr>
<td>HELD BY PT :</td>
<td>0</td>
</tr>
<tr>
<td>FREE PAGES :</td>
<td>0</td>
</tr>
<tr>
<td>FAILS DUE TO RDS POOL FULL :</td>
<td>0</td>
</tr>
<tr>
<td>XPROC REQUESTS :</td>
<td>13</td>
</tr>
<tr>
<td>XPROC ALLOC STOR :</td>
<td>N/A</td>
</tr>
<tr>
<td>PLAN BTB STORAGE :</td>
<td>0</td>
</tr>
<tr>
<td>PKG BTB STORAGE :</td>
<td>0</td>
</tr>
<tr>
<td>PLAN ATB STORAGE :</td>
<td>16896</td>
</tr>
<tr>
<td>PKG ATB STORAGE :</td>
<td>0</td>
</tr>
<tr>
<td>REQ STOR FOR STATIC STMTS:</td>
<td>21872</td>
</tr>
</tbody>
</table>

### PAGES IN POOL

This field shows the sum of the values for the following counters:

- HELD BY CT
- HELD BY PT
- FREE PAGES

Field Name: QISEPAGE

### CT PAGES

The current number of pages used for the cursor tables (CTs). This is a snapshot value.

Field Name: QISECT

### PT PAGES

The current number of pages used for package tables (PTs). This is a snapshot value.

Field Name: QISEKT

### FREE PAGES

The number of pages currently not used by any object in the EDM pool, in the EDM pool (below), or in the RDS pool (below). This is a snapshot value.

Field Name: QISEFREE

### CT REQUESTS

The number of requests for cursor table (CT) sections.

Field Name: QISECTG

### PT REQUESTS
IFCID 002 - EDM Pool Data

The number of requests for package table (PT) sections.
Field Name: QISEK TG

DBD REQUESTS
The number of requests for database descriptors (DBDs).
Field Name: QISED BDG

EDM POOL FULL
The total number of failures because the EDM pool or EDM pool (below) was full.
Field Name: QISEFAIL
This is an exception field.

CT NOT IN POOL
The number of times a cursor table section was loaded from DASD.
To find the number of times the CT was found in the EDM pool, subtract this value from the value of the Requests for sections - CT field.
Field Name: QISECTL
This is an exception field.

PT NOT IN POOL
The number of times a package table section was loaded from DASD.
To find the number of times the PT was already in the EDM pool, subtract this value from the value of the Requests for sections - PT field.
Field Name: QISEKTL
This is an exception field.

DBD NOT IN POOL
The total number of times database descriptors were loaded from DASD.
To find the number of times the DBD was already in the EDM pool, subtract this value from the value of Requests for sections - DBD field.
Field Name: QISED BDL
This is an exception field.

CACHE INSERTS
The number of full prepare requests.
A Full Prepare occurs for both Explicit Prepare and Implicit Prepare requests when the skeleton copy of the prepared SQL statement is not found in global dynamic SQL cache in the EDM pool.
Field Name: QISEDSI
This is an exception field.

CACHE REQUESTS
The number of requests for prepared statement cache sections.
Field Name: QISED SG

PKG SEARCH NOT FOUND
When a package is bound with a wild card (*) for package names, in the form of PKLIST(COL1.*,COL2.*....), EDM generates a NOT-FOUND record to avoid future I/O if a collection ID/package name combination does not exist.

This field shows how often a cached record was located during package binding.

Field Name: QISEKNFM

PKG SEARCH NOT FOUND INSERT

When a package is bound with a wild card (*) for package names, in the form of PKLIST(COL1.*,COL2.*....), EDM generates a NOT-FOUND record to avoid future I/O if a collection ID/package name combination does not exist.

This field shows how often a record was added to the cache during package binding.

Field Name: QISEKNFA

PKG SEARCH NOT FOUND DELETE

When a package is bound with a wild card (*) for package names, in the form of PKLIST(COL1.*,COL2.*....), EDM generates a NOT-FOUND record to avoid future I/O if a collection ID/package name combination does not exist.

This field shows how often a record was removed from the cache during package binding.

Field Name: QISEKNFR

STATEMENTS IN GLOBAL CACHE

Number of statements in the global cache.

Field Name: QISESTMT

PAGES IN STMT POOL (ABOVE)

The current number of pages in the EDM Statement pool above the 2 GB bar. This is a snapshot value.

Field Name: QISECPGE

PAGES IN DBD POOL (ABOVE)

This field shows the number of pages in the DBD pool above the 2 GB bar.

Field Name: QISEDTPGE

HELD BY STATEMENTS

The number of pages in the EDM Statement pool above the 2 GB bar that is used for cached dynamic SQL statements. This is a snapshot value.

Field Name: QISEDYNP

HELD BY DBD

The current number of pages used for database descriptors (DBDs). This is a snapshot value.

Field Name: QISEDDBD

FREE PAGES
IFCID 002 - EDM Pool Data

This field shows the number of free pages in the DBD pool above the 2 GB bar.
Field Name: QISEDFRE
This is an exception field.

STEALABLE PAGES
The current number of stealable pages used for skeleton cursor and package tables.
Field Name: QISEKLRU

FAILS DUE TO STMT POOL FULL
The total number of failures because the EDM Statement pool above the 2 GB bar was full.
Field Name: QISECFAL

FREE PAGES
The number of pages currently not used by any object in the EDM Statement pool above the 2 GB bar.
Field Name: QISECFRE

FAILS DUE TO DBD POOL FULL
This field shows the total number of failures because the DBD pool above the 2 GB bar was full.
Field Name: QISEDFAL
This is an exception field.

PAGES IN RDS POOL (ABOVE)
The number of pages in the RDS pool above the 2 GB bar.
Field Name: QISESPGE

PAGES IN SKEL POOL (ABOVE)
The current number of pages in the EDM skeleton pool above the 2 GB bar.
Field Name: QISEKPGE

HELD BY CT
The number of pages in the RDS pool above the 2 GB bar used for the cursor tables (CTs). This is a snapshot value.
Field Name: QISECTA

HELD BY SKCT
The current number of pages used for skeleton cursor tables (SKCTs). This is a snapshot value.
Field Name: QISESKCT

HELD BY PT
The number of pages in the RDS pool above the 2 GB bar used for the package tables (PTs). This is a snapshot value.
Field Name: QISEKTA
HELD BY SKPT
The current number of pages used for skeleton package tables (SKPTs).
This is a snapshot value.
Field Name: QISESKPT

FREE PAGES
The number of pages currently not used by any object in the EDM skeleton pool above the 2 GB bar.
Field Name: QISEKFRE

STEALABLE PAGES
The current number of stealable pages used for database descriptors (DBDs).
Field Name: QISEDLRU

FAILS DUE TO RDS POOL FULL
The number of failures because the RDS pool above the 2 GB bar was full.
Field Name: QISESFAL

FREE PAGES
The number of free pages in the RDS pool above the 2 GB bar.
Field Name: QISESFRE

FAILS DUE TO SKEL POOL FULL
The total number of failures because the EDM skeleton pool above the 2 GB bar was full.
Field Name: QISEKFAL

XPROC REQUESTS
The number of shareable static SQL statement requests. Prior to DB2 11, the field is used for the number of requests of executable code sequences (xPROC).
Field Name: QISEKSPG

XPROC ALLOC STOR
The total storage allocated to executable code sequences (xPROC).
Field Name: QISEKSPA

PLAN BTB STORAGE
The storage allocated to plans below the bar.
Field Name: QISESQCB

PKG BTB STORAGE
The storage allocated to packages below the bar.
Field Name: QISESQKB

PLAN ATB STORAGE
The storage allocated to plans above the bar
Field Name: QISESQCA
IFCID 002 - EDM Pool Data

PKG ATB STORAGE

The storage allocated to packages above the bar.

Field Name: QISESQKA

REQ STOR FOR STATIC STMTS

The total storage requested for shareable static SQL statements.

Field Name: QISEKSPA8
IFCID 002 - Group Buffer Pools Activity Data

This topic shows detailed information about “Record Trace - IFCID 002 - Group Buffer Pools Activity Data”.

Record trace - IFCID 002 - Group Buffer Pools Activity Data

The field labels shown in the following sample layout of “Record Trace - IFCID 002 - Group Buffer Pools Activity Data” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GROUP BUFFER POOL ID</td>
<td>The group buffer pool identifier.</td>
</tr>
<tr>
<td>FLAGS</td>
<td>The flag byte shows if more QBGL data is following or if this is the last of</td>
</tr>
<tr>
<td>SYN.READS(XI)-DATA RETURNED</td>
<td>the QBGL repeating groups.</td>
</tr>
<tr>
<td>SYN.READS(NF)-DATA RETURNED</td>
<td>The number of requests made to read a page from the group buffer pool</td>
</tr>
<tr>
<td>READ FOR CASTOUT MULT</td>
<td>because the page was invalidated in the member's buffer pool. The member</td>
</tr>
<tr>
<td>CLEAN PAGES WRITTEN</td>
<td>found the required page in the group buffer pool.</td>
</tr>
<tr>
<td>CHANGED PAGES SYNC.WRITTEN</td>
<td>The number of requests made to read a page from the group buffer pool</td>
</tr>
<tr>
<td>WRITE FAILED-NO STORAGE</td>
<td>because the page was not in the buffer pool of the member. The member found</td>
</tr>
<tr>
<td>REG.PAGE LIST (RPL) REQ.</td>
<td>the required page in the group buffer pool.</td>
</tr>
<tr>
<td>GBP CHECKPOINTS TRIGGERED</td>
<td>The number of requests made to read a page from the group buffer pool</td>
</tr>
<tr>
<td>PAGES RETRIEVED FROM GBP</td>
<td>because the page was not in the buffer pool of the member. The member found</td>
</tr>
<tr>
<td>READ STORAGE STATS</td>
<td>the required page in the group buffer pool.</td>
</tr>
<tr>
<td>DELETE NAME</td>
<td>This is an exception field.</td>
</tr>
<tr>
<td>UNREGISTER PAGE</td>
<td>The number of requests made to read a page from the group buffer pool</td>
</tr>
<tr>
<td>REGISTER PAGE</td>
<td>because the page was not in the buffer pool of the member. The member found</td>
</tr>
<tr>
<td>EXPLICIT X-INVALID</td>
<td>the required page in the group buffer pool.</td>
</tr>
<tr>
<td>PG P-LOCK UNLOCK REQ</td>
<td>The number of requests made to read a page from the group buffer pool</td>
</tr>
<tr>
<td>PG P-LOCK LOCK REQ SP MAP PG</td>
<td>because the page was not in the buffer pool of the member. The member found</td>
</tr>
</tbody>
</table>

Background and Tuning Information

When you increase the size of the group buffer pool (GBP), the number of pages returned from the GBP can increase. Conversely, decreasing the size of the GBP can cause DB2 to return fewer pages because the GBP cannot hold pages long enough to allow them to be retrieved again.

Field Name: QBGLXD

This is an exception field.

SYN.READS(NF)-DATA RETURNED

The number of requests made to read a page from the group buffer pool because the page was not in the buffer pool of the member. The member found the page in the group buffer pool.

Background and Tuning Information

The requesting member needs a page from a table space or index that is GBP-dependent or has GBPcache ALL defined. To get that page, the group buffer pool is checked before the page set on DASD.
If the group buffer pool is used to cache both clean and changed pages (GBPCACHE ALL is used for all data), you can try to get more pages returned from the group buffer pool by increasing the size of the group buffer pool. Do not tune the GBP based on this counter if it is used for caching changed pages only (GBPCACHE CHANGED).

**Field Name:** QBGLMD

This is an exception field.

**SYN.READS(XI)-NO DATA RET.**

The number of requests to read a page from the group buffer pool made because the page was invalidated in the member's buffer pool. The page was not found in the GBP and the page was recovered from DASD.

**Field Name:** QBGLGXR

**SYN.READS(NF)-NO DATA RET.**

The number of requests made to read a page from the group buffer pool because the page was not in the member's buffer pool. The member did not find the required data in the group buffer pool and had to retrieve the page from DASD.

**Background and Tuning Information**

The requesting member needs a page from a table space or index that is GBP-dependent or has GBPCACHE ALL defined. To get that page, the group buffer pool is checked before the page set on DASD.

You can compare the value in this counter with the number of pages that were returned from the group buffer pool, see Sync.Read (Not Found) - Data Returned. If the group buffer pool is used to cache both clean and changed pages (GBPCACHE ALL is used for all data), you can try to get more pages returned from the group buffer pool by increasing the size of the group buffer pool. Do not tune the GBP based on this counter if it is used for caching changed pages only (GBPCACHE CHANGED).

**Field Name:** QBGLMR

**READ FOR CASTOUT MULT**

The number of Read For Castout Multiple requests.

**Field Name:** QBGLCM

**WRITE AND REGISTER MULT**

The number of Write and Register Multiple requests.

**Field Name:** QBGLWM

**READ FOR CASTOUT**

The number of Read For Castout requests. One page read per request.

**Field Name:** QBGLCR

**WRITE AND REGISTER**

The number of Write and Register requests.

**Field Name:** QBGLWS

**CLEAN PAGES WRITTEN**
The number of clean pages that were synchronously written to the group buffer pool from the virtual pool.

**Background and Tuning Information**

Only GBPCACHE ALL causes clean (unchanged) pages to be written to the coupling facility. The pages are written to the coupling facility even if the page set is not GBP-dependent. If group buffer pool caching works effectively for prefetch, the value in this field should be much smaller than the value in Synchronous Read (Not Found) - Data Returned.

**Field Name:** QBGLWC

This is an exception field.

**PAGES WRITE & REG MULT**

The number of pages written using Write and Register Multiple (WARM) requests.

**Field Name:** QBGLWP

**CHANGED PAGES SYNC.WRITTEN**

The number of changed pages written synchronously to the group buffer pool.

Pages are written with Write and Register (WAR) requests or Write and Register Multiple (WARM) requests.

At commit time changed pages are forced from the virtual buffer pool of the member to the coupling facility.

**Background and Tuning Information**

In data sharing, changed pages must have been written to the group buffer pool by the time a transaction commits. The pages are written either synchronously (force at commit) or asynchronously, for example, when a local buffer pool threshold is reached or at a member's checkpoint. The number of pages that have to be forced out synchronously (in “burst mode”) at commit time can be reduced if asynchronous writes are triggered more frequently.

You can use the vertical deferred write threshold (VDWQT) to reduce the number of pages that have to be forced out synchronously and to increase the number of pages that are asynchronously written before the transaction commits. For GBP-dependent page sets, writes triggered by the vertical deferred write threshold go to the coupling facility. You can cause changed pages to be written out quicker and in smaller increments, by reducing the vertical deferred write threshold (VDWQT).

**Field Name:** QBGLSW

This is an exception field.

**PAGES CASTOUT**

The number of data pages that were cast out of the group buffer pool of the member.

Castout to a page set or partition is done by the castout owner of the page set or partition. This is normally the DB2 subsystem that had the first update intent on the page set or partition.

**Background and Tuning Information**
The number of pages written per I/O is normally close to the value of this field divided by the value in Unlock Castout.

For example, if an average of four pages is written per castout write I/O, the number of pages cast out should be four times the number in this field.

Because DB2 usually includes more than one page in the request to write pages to DASD, the number in this field should always be significantly more than Unlock Castout. If it is not (for example, when "unlock castout" is more than half of "pages castout"), the castout write I/O is inefficient; probably because you have random update patterns on the DB2 data or a low castout threshold.

**Field Name:** QBGLRC

This is an **exception** field.

**CHANGED PAGES ASYNC.WRITTEN**

The number of changed pages written asynchronously to the group buffer pool.

Pages are written in response to Write and Register (WAR) and Write and Register Multiple (WARM) requests.

Changed pages can be written from the member's virtual buffer pool to the group coupling facility before the application commits. This happens when, for example, a local buffer pool threshold is reached, or when P-lock negotiation forces the pages on the vertical deferred write queue to be written to the group buffer pool.

**Background and Tuning Information**

In data sharing, changed pages must have been written to the group buffer pool before a transaction commits. The pages are written either synchronously during commit processing or asynchronously before the transaction commits when, for example, a local buffer pool threshold is reached or at a member's checkpoint. See Changed Pages - Written Synchronously for the number of changed pages synchronously written to the group buffer pool.

The vertical deferred write threshold (VDWQT) can be used to reduce the number of pages that have to be forced out synchronously and to increase the number of pages that are asynchronously written before the transaction commits. For GBP-dependent page sets, writes triggered by the vertical deferred write threshold go to the coupling facility. If you want changed pages to be written out quicker and in smaller increments, you can lower the vertical deferred write threshold (VDWQT).

**Field Name:** QBGLAW

This is an **exception** field.

**CASTOUT CLASS THRESHOLD**

The number of times group buffer pool castout was initiated because the group buffer pool class castout threshold was detected.

**Background and Tuning Information**

The class castout threshold is one of two group buffer pool thresholds. In most cases the default value for the class threshold (10 percent) is a good choice. Depending on your workload, altering this value can reduce DASD contention during castout.
Field Name: QBGLCT
This is an exception field.

WRITE FAILED-NO STORAGE
The number of coupling facility write requests that could not complete due to a lack of coupling facility storage resources.

Background and Tuning Information
A value greater than zero indicates that the data page resources of the coupling facility are being consumed faster than the DB2 castout processes can free them.

On write failure, the affected DB2 member initiates castout and retries several times, and finally, if it is a changed page, it will be added to the logical page list (LPL) requiring recovery.

If the problem is not simply due to a momentary surge in activity, you need either to decrease the group buffer pool castout thresholds, or to increase the number of data entries in the group buffer pool. To increase the number of data entries, you can do one of the following:

- Increase the total size of the group buffer pool.
- Adjust the ratio of directory entries to data entries in favor of data entries.

Field Name: QBGLWF
This is an exception field.

GROUP BP CASTOUT THRESHOLD
The number of times a group buffer pool castout was initiated because the group buffer pool castout threshold was detected.

Background and Tuning Information
The GBP castout threshold, GBP class castout threshold, and the length of the GBP checkpoint interval determine the castout characteristics of the group buffer pool.

You can consider this threshold a safety margin to protect the group buffer pool from being accidentally flooded by overactive applications.

In most situations, the default value for the group buffer pool castout threshold of 50 percent is a good choice. Use the ALTER GROUPBUFFERPOOL command to tune the group buffer pool thresholds.

Field Name: QBGLGT
This is an exception field.

REG.PAGE LIST (RPL) REQ.
The number of register page list (RPL) requests made by prefetch. The group buffer pool must be allocated in a group coupling facility with CFLEVEL=2 or higher.

Background and Tuning Information
Performance might be improved by enabling RPL.

Field Name: QBGLAX
This is an exception field.
IFCID 002 - Group Buffer Pools Activity Data

DELETE NAME LIST SEC-GBP
The number of DELETE NAME LIST requests to delete pages from the secondary group buffer pool that have just been cast out from the primary.
Field Name: QBGL2D
This is an exception field.

GBP CHECKPOINTS TRIGGERED
The number of group buffer pool checkpoints triggered by this member.

Background and Tuning Information
The value of this counter depends on the length of the group buffer pool checkpoint interval.
Field Name: QBGLCK

DELETE PAGE FROM SEC-GBP
The number of group buffer pool requests to delete a page from the secondary group buffer pool. These requests are issued by the group buffer pool structure owner to delete orphaned data entries in the secondary GBP as part of the garbage collection logic.
Field Name: QBGL2N

PAGES RETRIEVED FROM GBP
The number of coupling facility reads performed by prefetch to retrieve a changed page from the group buffer pool.
Field Name: QBGLAY
This is an exception field.

READ CASTOUT STATS SEC-GBP
The number of coupling facility requests to read the castout statistics for the secondary group buffer pool. These requests are issued by the group buffer pool structure owner to check for orphaned data entries in the secondary group buffer pool.
Field Name: QBGL2R
This is an exception field.

READ STORAGE STATS
The number of times DB2 requested statistics information from the group buffer pool. It is issued by the group buffer pool structure owner at timed intervals to determine whether the group buffer pool castout threshold (GBPOOLT) has been reached.
Field Name: QBGLOS

UNLOCK CASTOUT
The number of times DB2 issued an unlock request to the coupling facility for completed castout I/Os.
When pages are cast out to DASD, they are locked for castout in the coupling facility. This castout lock is not an IRLM lock; it is to ensure that only one system can cast out a given page at a time.

Background and Tuning Information
The number of pages written per I/O is normally close to the value of pages castout divided by the value of this field.

For example, if an average of four pages is written per castout write I/O, the number of pages cast out should be four times the value in this field. Because DB2 usually includes more than one page in a write request, the number in this field should always be significantly less than pages castout. If it is not (for example, when "unlock castout" is more than half of "pages castout"), the castout write I/O is inefficient; possibly because you have random update patterns on the DB2 data or a low castout threshold.

Field Name: QBGLUN

DELETE NAME

The number of requests made by DB2 to delete directory and data entries associated with a particular page set or partition from the group buffer pool.

DB2 issues this request when it changes a page set or partition from GBP-dependent to non GBP-dependent. DB2 also issues this request for objects that are defined with GBPCACHE ALL when those objects are first opened.

Background and Tuning Information

This counter is a measure of how often page sets or partitions change between being and not being dependent on the group buffer pool.

You can prevent DB2 going in and out of GBP dependency too often by tuning the following subsystem parameters that affect data sets when they are switched to a different state:

PCLOSEN

Pseudoclose frequency. The number of checkpoints required before a data set that was not updated can be a pseudoclose candidate.

If the PCLOSEN condition is met, the page set or partition is converted from read-write to read-only state. Depending on other concurrent users, this could raise the chance for the page set or partition to go out of GBP dependency.

PCLOSET

Pseudoclose time. The amount of time (in minutes) that must elapse before a data set can be a pseudoclose candidate.

If the PCLOSEN or PCLOSET condition is met, the page set or partition is converted from read-write to read-only state. Depending on other concurrent users, this could raise the chance for the page set or partition to go out of GBP dependency.

LOGLOAD

The number of log records that DB2 writes between successive checkpoints.

These parameters are specified in the CHECKPOINT FREQ field in panel DSNTPN.

Field Name: QBGLDN

READ CASTOUT CLASS
The number of requests made to the group buffer pool to determine which pages, from a particular page set or partition, must be cast out because they are cached as changed pages.

This request is issued either by the page set or partition castout owner, or, when the group buffer pool castout threshold is reached, by the group buffer pool structure owner.

Field Name: QBGLCC

UNREGISTER PAGE

The number of times DB2 unregistered interest for a single page. This happens when DB2 steals pages from the member's buffer pool that belong to GBP-dependent page sets or partitions.

Background and Tuning Information

A large value here indicates that the local buffer pool contains a mixture of GBP-dependent data and non-GBP-dependent data.

The page stolen from the local buffer pool is replaced by a new one. This counter makes a distinction on whether the new page depends on the group buffer pool or not.

Usually a page of a GBP-dependent page set or partition is replaced by a page that is also GBP-dependent. In this instance, the unregister request for the page being stolen is combined with the read and register request for the new page. These combined requests do not contribute to this counter.

If, however, a page of a GBP-dependent page set or partition is replaced by a page that is not GBP-dependent, then only an unregister request is sent to the coupling facility. These separate requests are counted here.

Field Name: QBGLDG

NR.OF READ FOR CASTOUT REQ.

The number of requests issued by the group buffer pool structure owner to determine which castout classes have changed pages.

This request is made by the group buffer pool structure owner when the group buffer pool threshold is reached. Normally, you would expect only one or two requests each time the group buffer pool threshold is reached.

Field Name: QBGLCS

REGISTER PAGE

The number of times DB2 registered interest in a single page.

These are "register-only" requests, which means that DB2 is not requesting any data back from the request.

This request is made only to create a directory entry for the page to be used for cross-invalidation when the page set or partition P-lock is downgraded from S to IS mode, or from SIX to IX mode.

Field Name: QBGLRG

READ DIRECTORY INFO

The number of requests issued by the group buffer pool structure owner to read the directory entries of all changed pages in the group buffer pool.
This request is issued at group buffer pool checkpoints to record the oldest recovery log record sequence number (LRSN). It is used as a basis for recovery if the group buffer pool fails.

Such requests might have to be issued several times for each group buffer pool checkpoint to read the directory entries for all changed pages.

**Background and Tuning Information**

If the value of this counter appears to be abnormally high, consider upgrading the coupling facility to CFLEVEL=2 or higher to raise the number of directory entries that can be read with one request. You can also increase the group buffer pool checkpoint interval, but this can lengthen the recovery for the group buffer pool.

**Field Name:** QBGLRD

**EXPLICIT X-INVALID**

The number of times an explicit coupling facility cross-invalidation request was issued.

**Field Name:** QBGLEX

**GBP-DEPENDENT GETPAGES**

The number of Getpages made for GBP-dependent objects.

**Field Name:** QBGLGG

**PG P-LOCK UNLOCK REQ**

The number of page P-lock unlock requests.

**Field Name:** QBGLU1

**ASYNCH GBP REQUESTS**

The number of asynchronous IXLCACHE invocations for the primary group buffer pool.

**Field Name:** QBGAHS

**PG P-LOCK LOCK REQ SP MAP PG**

The number of page P-lock lock requests for space map pages.

**Field Name:** QBGLP1

**ASYNCH SEC-GBP REQUESTS**

The number of IXLCACHE invocations for the secondary group buffer pool.

**Field Name:** QBGA2H
IFCID 002 - Locking Data

This topic shows detailed information about “Record Trace - IFCID 002 - Locking Data”.

Record trace - IFCID 002 - Locking Data

The field labels shown in the following sample layout of “Record Trace - IFCID 002 - Locking Data” are described in the following section.

DEADLOCKS

The number of times deadlocks were detected. This number should be low, ideally 0.

Background and Tuning Information

Deadlocks occur when two or more application processes each hold locks on resources that the others need, without which they cannot proceed. Ensure that all applications accessing the same tables access them in the same order.

Deadlocks can also occur through index page splits if there is high insert activity. In this case, the recommendation is to set SUBPAGES to 1 for the index.

This field is incremented once for each deadlock encountered. There is no correlation between this field and the deadlock events reported in the Locking report set or the number of IFCID 172 records written. This field reports all deadlocks, regardless of how they were resolved. The locking report and record trace IFCID 172 show only those deadlocks that were resolved by DB2.

Field Name: QTXADEA

This is an exception field.

LOCK REQUEST

The number of requests to lock a resource.

Field Name: QTXALOCK

This is an exception field.

LOCK SUSPENSIONS

The number of times a lock could not be obtained and the unit of work was suspended.

Background and Tuning Information

This number should be low, ideally 0.

The number of lock suspensions is a function of the lock requests. Lock suspensions (or conflicts) can happen on either LOCK REQUEST or CHANGE REQUEST.

Suspensions are highly dependent on the application and table space locking protocols.

Field Name: QTXASLOC
CLAIM REQUESTS
The number of claim requests.
Field Name: QTXACLNO
This is an exception field.

TIMEOUTS
The number of times a unit of work was suspended for a time exceeding the timeout value. This number should be low, ideally 0.
Field Name: QTXATIM
This is an exception field.

UNLOCK REQUEST
The number of requests to unlock a resource.
This value can be less than the number of lock requests because DB2 can release several locks with a single unlock request.
Field Name: QTXAUNLK

IRLM LATCH SUSPENS.
The number of latch suspensions.
Field Name: QTXASLAT

CLAIM REQ. FAILED
The number of unsuccessful claim requests.
Field Name: QTXACLUN

ESCALATIONS(SHR)
The number of times the maximum page locks per table space are exceeded, and the table space lock escalates from a page lock (IS) to a table space lock (S) for this thread. You can specify the number of locks allowed per table space with the LOCKS PER TABLE(SPACE) parameter on the DB2 install panel DSNTIPJ.

Background and Tuning Information
Escalations can cause unpredictable response times. Lock escalations should only happen when an application process updates or references (if repeatable read is used) more pages than normal.
Field Name: QTXALES
This is an exception field.

QUERY REQUEST
The number of query requests.
Field Name: QTXAQRY
This is an exception field.

OTHER SUSPENSIONS
The number of suspensions caused by something other than lock or latch.
Field Name: QTXASOTH
IFCID 002 - Locking Data

This is an exception field.

DRAIN REQUESTS

The number of drain requests.

Field Name: QTXADRNO

This is an exception field.

ESCALATIONS (EXC)

The number of times the maximum page locks per table space are exceeded and the table space lock escalates from a page lock (IX) to a table space lock (X).

Background and Tuning Information

Escalations can cause unpredictable response times. Lock escalations should only happen when an application process updates or references (if repeatable read is used) more pages than it normally does.

A useful rule of thumb is to compare the number of escalations (shared and exclusive) to the successful escalations (those that did not cause deadlocks and timeouts). If this value, or the number Lock escalations - shared and if the number of timeouts or deadlocks is also not 0, the timeout or deadlock is probably caused by the escalation.

If many escalations cause deadlocks and timeouts, the recommendation is to change the escalation threshold value. Use of ANY is extremely useful to prevent unnecessary and expensive page locks, for example locking all pages in a table space.

Lock escalations, shared or exclusive, should not be expected in a transaction environment.

Field Name: QTXALEX

This is an exception field.

CHANGE REQUEST

The number of change requests.

Field Name: QTXACHG

This is an exception field.

DRAIN REQ. FAILED

The number of unsuccessful drain requests.

Field Name: QTXADRUN

This is an exception field.

MAXIMUM PAGE/ROW LOCKS HELD

The maximum number of page or row locks concurrently held against all table spaces by a single application during its execution. This count is a high-water mark. It cannot exceed the LOCKS PER USER parameter on panel DSNTIPJ.

Field Name: QTXANPL

This is an exception field.

OTHER REQUEST
IFCID 002 - Locking Data

The number of requests to IRLM to perform a function other than LOCK, UNLOCK, QUERY, or CHANGE.

Field Name: QTXAIRLM

This is an exception field.
IFCID 002 - Miscellaneous

IFCID 002 - Miscellaneous
This topic shows detailed information about “Record Trace - IFCID 002 - Miscellaneous”.

Record trace - IFCID 002 - Miscellaneous
The field labels shown in the following sample layout of “Record Trace - IFCID 002 - Miscellaneous” are described in the following section.

<table>
<thead>
<tr>
<th>MISCELLANEOUS</th>
<th>MX ST LOB (MB)</th>
<th>MX ST XML (MB)</th>
<th>0</th>
<th>ARRAY EXPANSIONS</th>
<th>0</th>
<th>SPARSE IX DISABLED</th>
<th>0</th>
<th>SPARSE IX BUILT WF</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>QXPFSLNUM ........:</td>
<td>0</td>
<td>QXPFSENUM ........:</td>
<td>0</td>
<td>QXPFSENUMG ........:</td>
<td>0</td>
<td>QXN1093A ........:</td>
<td>0</td>
<td>QXN1093B ........:</td>
<td>0</td>
</tr>
<tr>
<td>QXPFMAXU ........:</td>
<td>0</td>
<td>QXPFMAXUG ........:</td>
<td>0</td>
<td>QXN1093A ........:</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

MX ST LOB VAL (MB)
Maximum storage used for LOB values.
Field Name: QXSTLOBV

MX ST XML VAL (MB)
Maximum storage used for XML values.
Field Name: QXSTXMLV

ARRAY EXPANSIONS
The number of times an array variable is expanded beyond 32 KB.
Field Name: QXSTARRAY_EXPANSIONS

SPARSE IX DISABLED
The number of times that sparse index was disabled because of insufficient storage.
Field Name: QXSISTOR

SPARSE IX BUILT WF
The number of times that sparse-index built a physical work file for probing.
Field Name: QXSIWF

QXPFSNUM
This field is for IBM service use.
Field Name: QXPFSNUM

QXPFSNUMG
This field is for IBM service use.
Field Name: QXPFSNUMG

QXPFMAXU
This field is for IBM service use.
Field Name: QXPFMAXU
### IFCID 002 - Miscellaneous

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>QXPFMAXUG</td>
<td>This field is for IBM service use.</td>
</tr>
<tr>
<td>Field Name: QXPFMAXUG</td>
<td></td>
</tr>
<tr>
<td>QXN1093A</td>
<td>This field is for IBM service.</td>
</tr>
<tr>
<td>Field Name: QXN1093A</td>
<td></td>
</tr>
<tr>
<td>QXN1093B</td>
<td>This field is for IBM service.</td>
</tr>
<tr>
<td>Field Name: QXN1093B</td>
<td></td>
</tr>
</tbody>
</table>
IFCID 002 - Nested SQL Activity

IFCID 002 - Nested SQL Activity

This topic shows detailed information about “Record Trace - IFCID 002 - Nested SQL Activity”.

Record trace - IFCID 002 - Nested SQL Activity

The field labels shown in the following sample layout of “Record Trace - IFCID 002 - Nested SQL Activity” are described in the following section.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAX CASCAD LVL</td>
<td>The maximum level of indirect SQL cascading. This includes cascading because of triggers, UDFs, or stored procedures.</td>
</tr>
<tr>
<td>Field Name: QXCASCDP</td>
<td>This is an exception field.</td>
</tr>
<tr>
<td>CALL STATEMENTS</td>
<td>The number of SQL CALL statements executed.</td>
</tr>
<tr>
<td>Field Name: QXCALL</td>
<td>This is an exception field.</td>
</tr>
<tr>
<td>UDF EXECUTED</td>
<td>The number of user-defined functions executed.</td>
</tr>
<tr>
<td>Field Name: QXCAUD</td>
<td>This is an exception field.</td>
</tr>
<tr>
<td>UDF ABENDS</td>
<td>The number of times a user-defined function abended.</td>
</tr>
<tr>
<td>Field Name: QXCAUDAB</td>
<td>This is an exception field.</td>
</tr>
</tbody>
</table>

MAX CASCAD LVL

The maximum level of indirect SQL cascading. This includes cascading because of triggers, UDFs, or stored procedures.

Field Name: QXCASCDP
This is an exception field.

CALL STATEMENTS

The number of SQL CALL statements executed.

Field Name: QXCALL
This is an exception field.

PROCEDURE ABENDS

The number of times a stored procedure terminated abnormally.

Field Name: QXCALLAB
This is an exception field.

CALL TIMEOUTS

The number of times an SQL call timed out waiting to be scheduled.

Field Name: QXCALLTO
This is an exception field.

CALL REJECTS

The number of times an SQL CALL statement was rejected due to the procedure being in the STOP ACTION(REJECT) state.

Field Name: QXCALLRJ
This is an exception field.

UDF EXECUTED

The number of user-defined functions executed.

Field Name: QXCAUD
This is an exception field.

UDF ABENDS

The number of times a user-defined function abended.

Field Name: QXCAUDAB
This is an exception field.
UDF TIMEOUTS
The number of times a user-defined function timed out while waiting to be scheduled.

Field Name: QXCAUDTO
This is an exception field.

UDF REJECTS
The number of times a user-defined function was rejected.

Field Name: QXCAUDRJ
This is an exception field.

STMT TRIGGER
The number of times a statement trigger was activated.

Field Name: QXSTTRG
This is an exception field.

ROW TRIGGER
The number of times a row trigger was activated.

Field Name: QXROWTRG
This is an exception field.

SQL ERROR TRIGGER
The number of times an SQL error occurred during the execution of a triggered action. This includes errors that occur in user-defined functions or stored procedures that are called from triggers and that pass back a negative SQLCODE.

Field Name: QXTRGERR
This is an exception field.
IFCID 002 - Query Parallelism

**IFCID 002 - Query Parallelism**

This topic shows detailed information about “Record Trace - IFCID 002 - Query Parallelism”.

**Record trace - IFCID 002 - Query Parallelism**

The field labels shown in the following sample layout of “Record Trace - IFCID 002 - Query Parallelism” are described in the following section.

```
  QUERY PARALLELISM

MAX DEG ESTIMATED: 0 MEMBERS SKIPPED : 0 PARALL. DISABLED : N/A
MAX DEG PLANNED : 0
MAX DEG EXECUTED : 0

Parallel Groups:
FALL TO SEQ-CURSOR 0 GROUPS EXECUTED : 0 ONE DB2-COORD=NO : 0
FALL TO SEQ-NOESA: 0 RAN REDUCED-STORE : 0 ONE DB2-ILO LVL : 0
FALL TO SEQ-NEGOTN 0 RAN REDUCED-NEGOTN 0 REFORM PARAL-CFG : 0
FALL TO SEQ-ENCLV: N/A 0
FALL TO SEQ-PROOF 0 RAN AS PLANNED ...: 0 REFORM PARAL-BUF : 0
FALL TO SEQ-ENV: N/A
QXPAROPT ..........: 0
```

**MAX DEG ESTIMATED**

The maximum degree of parallelism estimated for a parallel group at bind time based on the cost formula. If the parallel group contains a host variable or parameter marker, then bind time will estimate the parallel group degree based on a valid assumption value.

**Field Name:** QXMAXESTIDG

**MEMBERS SKIPPED**

The number of times the parallelism coordinator had to bypass a DB2 when distributing tasks because one or more DB2 members did not have enough buffer pool storage. The number in this field is only incremented at the parallelism coordinator once per parallel group, even though more than one DB2 might have lacked buffer pool storage for that parallel group. It is also only incremented when the buffer pool is defined to allow for parallelism. For example, if VPXPSEQT=0 on an assistant, DB2 does not send parallel work there and the number in this field is not incremented.

**Field Name:** QXXCSKIP

**PARALL. DISABLED**

Indicates whether query parallelism is disabled by the Resource Limit Facility for at least one dynamic select statement in this thread. A non-zero value means that query parallelism is disabled.

**Field Name:** QXRLFDPA

**MAX DEG PLANNED**

The maximum degree of parallelism planned for a parallel group. It is the ideal parallel group degree obtained at execution time after the host variable or parameter marker value is “plug-in” and before buffer pool negotiation and system negotiation are performed.

**Field Name:** QXMAXPLANIDG

**MAX DEG EXECUTED**

The maximum degree of parallelism executed among all parallel groups to indicate the extent to which queries were processed in parallel.

**Field Name:** QXMAXDEG
FALL TO SEQ-CURSOR
The total number of parallel groups that fell back to sequential mode due to a cursor that can be used by UPDATE or DELETE.

Field Name: QXDEGCUR

GROUPS EXECUTED
The total number of parallel groups executed.

Field Name: QXTOTGRP

ONE DB2-COORD=NO
The total number of parallel groups executed on a single DB2 subsystem due to the COORDINATOR subsystem value being set to NO. When the statement was bound, the COORDINATOR subsystem value was set to YES. This situation can also occur when a package or plan is bound on a DB2 subsystem with COORDINATOR=YES, but is run on a DB2 subsystem with COORDINATOR=NO.

Field Name: QXCOORNO

FALL TO SEQ-NOESA
The total number of parallel groups that fell back to sequential mode due to a lack of ESA sort support.

Field Name: QXDEGESA

PARALL.GROUPS
The total number of parallel groups that DB2 intended to run across the data sharing group. This number is only incremented at the parallelism coordinator at run time.

Field Name: QXXCBPNX

ONE DB2-ISO LVL
The total number of parallel groups executed on a single DB2 subsystem due to repeatable-read or read-stability isolation.

Field Name: QXISORR

FALL TO SEQ-STOR
The total number of parallel groups that fell back to sequential mode due to a storage shortage or contention on the buffer pool.

The exception field name is QXDEGBUF.

Field Name: QXDEGBUF

RAN REDUCED-STOR
The total number of parallel groups that did not reach the planned parallel degree because of a lack of storage space or contention on the buffer pool.

The exception field name is QXREDGRP.

Background and Tuning Information
If this field is not 0, increase the size of the current buffer pool using the ALTER BUFFERPOOL command or use the ALTER TABLESPACE command to assign table spaces accessed by this query to a different buffer pool.
**Field Name:** QXREDGRP

This is an *exception* field.

**ONE DB2-DCL TTAB**

The number of parallel groups in a query block that were downgraded to CPU parallelism because they referenced a UDF and a declared temporary table was detected at execution time.

DB2 enforces execution on a single DB2 (CPU parallelism), in this instance, because it cannot determine at incremental bind time for the statement whether the UDF will reference the declared temporary table. Other parallel groups in the same statement are not necessarily downgraded.

**Field Name:** QXDEGDTT

**FALL TO SEQ-NEGOTN**

The total number of parallel groups that fell back to sequential mode due to system negotiation result of system stress level.

**Field Name:** QXSTODGNGRP

**RAN REDUCED-NEGOTN**

The total number of parallel groups that did not reach the planned parallel degree due to system negotiation result of system stress level.

**Field Name:** QXSTOREDGRP

**REFORM PARAL-CFG**

The total number of parallel groups where DB2 reformulated the parallel portion of the access path because of a change in the number of active members, or because of a change of processor models on which they run, from bind time to run time. This counter is incremented only on the parallelism coordinator at run time.

**Field Name:** QXREPOP1

**FALL TO SEQ-A.PROC**

The total number of parallel groups that fell back to sequential mode under an autonomous procedure.

**Field Name:** QXDEGAT

**RAN AS PLANNED**

The total number of parallel groups that executed in the planned parallel degree. This field is incremented by one for each parallel group that executed in the planned degree of parallelism (as determined by DB2).

**Field Name:** QXNORGRP

**REFORM PARAL-BUF**

The total number of parallel groups in which DB2 reformulated the parallel portion of the access path because there were insufficient buffer-pool resources. This counter is incremented only at the parallelism coordinator at run time.

**Field Name:** QXREPOP2

**FALL TO SEQ-ENCLV**
IFCID 002 - Query Parallelism

The total number of parallel groups that executed in sequential mode due to the unavailability of MVS/ESA enclave services.

Field Name: QXDEGENC
This is an exception field.

QXPAROPT
This field is for IBM service.
Field Name: QXPAROPT
IFCID 002 - RID List Processing

IFCID 002 - RID List Processing
This topic shows detailed information about “Record Trace - IFCID 002 - RID List Processing”.

Record trace - IFCID 002 - RID List Processing

The field labels shown in the following sample layout of “Record Trace - IFCID 002 - RID List Processing” are described in the following section.

RL PROCESSING

<table>
<thead>
<tr>
<th>Field Label</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>RL PROCESSING USED</td>
<td>73361</td>
</tr>
<tr>
<td>RL PROCESSING NOT USED-NO STORAGE</td>
<td>11610</td>
</tr>
<tr>
<td>RL PROCESSING NOT USED-LIMIT EXCEEDED</td>
<td>0</td>
</tr>
<tr>
<td>RL OVERFLOWED-NO STORAGE</td>
<td>0</td>
</tr>
<tr>
<td>RL OVERFLOWED-MAX LIMIT</td>
<td>0</td>
</tr>
</tbody>
</table>

RL PROCESSING USED

The number of times RID list (also called RID pool) processing is used. During RID (RECORD ID) list processing, DB2 uses an index to produce a list of candidate RIDs, which is called a RID list. The RID list can be sorted and intersected (ANDED) or unioned (ORED) with other RID lists before actually accessing the data pages. RID list processing is used for a single index (index access with list prefetch) or for multiple indexes (multiple index access), which is when the RID lists are ANDED and ORED.

This field is incremented once for a given table access when RID list processing is used for index access with list prefetch, for multiple index access, or for both. For multiple index access, if a final RID list is obtained through ANDing and ORing of RID lists, the counter is incremented once, even if not all indexes were used by the RIDs in the multiple index access.

Background and Tuning Information

A nonzero value in this field indicates that DB2 has used list prefetch. If this is the case, check the access path selection.

Field Name: QXMIAP
This is an exception field.

RL PROCESSING NOT USED-NO STORAGE

The number of times DB2 detected that no storage was available to hold a list of RIDs during a given RID list process involving one index (single index access with list prefetch) or multiple indexes (multiple index access).

This field can be incremented during retrieval, sorting, ANDing, and ORing of RID lists for index access with list prefetch (single index). For single index access, this field can only be incremented once per access. For multiple index access, it can be incremented for every index involved in the ANDing and ORing of RID lists.

Field Name: QXNSMIAP
This is an exception field.

RL PROCESSING NOT USED-LIMIT EXCEEDED

The number of times DB2 detected that a RID list exceeded one or more internal limits during a given RID list (or RID pool) process involving one index (single index access with list prefetch) or multiple indexes (multiple index access). The internal limits include the physical limitation of the number of RIDs a RID list can hold and threshold values for the retrieval, ORing, and ANDing of RIDs.
For index access with list prefetch (single index), this field can only be incremented during RID list retrieval. For multiple index access, this field can be incremented during RID list retrieval, ANDing, and ORing. This counter reflects the number of times internal limits or threshold values were exceeded for the RID lists obtained directly from an index as well as for RID lists derived during the ANDing and ORing process.

**Background and Tuning Information**

Before you increase the RID list storage size, investigate the cause of the failure using the statistics record or the performance trace. You can specify the desired size for the RID list (within the range of 16 KB to 1000 MB) on the DB2 installation panel DSNTIPC.

**Field Name:** QXMRMIAP

This is an *exception* field.

**RL SKIPPED-INDEX KNOWN**

The number of times a RID list retrieval for multiple index access was skipped because it was not necessary due to DB2 being able to predetermine the outcome of index ANDing or ORing.

**Field Name:** QXRSMIAP

**RL OVERFLOWED-NO STORAGE**

The number of times a RID list was overflowed to a work file because no RID pool storage was available to hold the list of RIDs.

**Field Name:** QXWFRIDS

**RL INTERRUPTED-NO STORAGE**

The number of times a RID list append for a hybrid join was interrupted because no RID pool storage was available to hold the list of RIDs.

**Field Name:** QXHJINCS

**RL OVERFLOWED-MAX LIMIT**

The number of times a RID list was overflowed to a work file because the number of RIDs exceeded one or more internal limits.

**Field Name:** QXWFRIDT

**RL INTERRUPTED-MAX LIMIT**

The number of times a RID list append for a hybrid join was interrupted because the number of RIDs exceeded one or more internal limits.

**Field Name:** QXHJINCT
IFCID 002 - ROWID

This topic shows detailed information about “Record Trace - IFCID 002 - ROWID”.

Record trace - IFCID 002 - ROWID

The field labels shown in the following sample layout of “Record Trace - IFCID 002 - ROWID” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DIRECT ACCESS</td>
<td>The number of times that direct row access was successful.</td>
</tr>
<tr>
<td>INDEX USED</td>
<td>The number of times an index was used to find a record.</td>
</tr>
<tr>
<td>TABLE SPACE SCAN USED</td>
<td>The number of times that an attempt to use direct row access reverted to using a table-space scan because DB2 was unable to use a matching index scan.</td>
</tr>
</tbody>
</table>

Background and Tuning Information

Ideally, this value should be 0.

Table-space scans can happen, for example, when a REORG is performed between the read of the ROWID column and the use of the host variable in the WHERE clause of the SQL statement. This causes the RID value in the host variable to be incorrect. DB2 first tries a matching-index scan before using a table-space scan.

To avoid table space scans, you can force the access path of an unsuccessful direct row access to use a matching index scan on the primary-index key by adding PKCOL to the WHERE clause in the SQL statement. .... WHERE ROWIDCOL=:HVROWID AND PKCOL=:HPK

Field Name: QXROITS
**IFCID 002 - Service Controller Data**

This topic shows detailed information about “Record Trace - IFCID 002 - Service Controller Data”.

**Record trace - IFCID 002 - Service Controller Data**

The field labels shown in the following sample layout of “Record Trace - IFCID 002 - Service Controller Data” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SERVICE CONTROLLER DATA</td>
<td></td>
</tr>
<tr>
<td>PLAN ALLOC.ATTMP</td>
<td>142</td>
</tr>
<tr>
<td>PACK.ALLOC.ATTMP</td>
<td>6</td>
</tr>
<tr>
<td>OPEN DATASETS - HWM</td>
<td>102</td>
</tr>
<tr>
<td>AUTHORIZ.ATTEMPTS</td>
<td>1034</td>
</tr>
<tr>
<td>PLAN ALLOC.SUCC.</td>
<td>142</td>
</tr>
<tr>
<td>PACK.ALLOC.SUCC.</td>
<td>6</td>
</tr>
<tr>
<td>OPEN DATASETS - CURR.</td>
<td>101</td>
</tr>
<tr>
<td>AUTHORIZ.SUCC-NO CAT.</td>
<td>37</td>
</tr>
<tr>
<td>PLANS BOUND</td>
<td>30</td>
</tr>
<tr>
<td>PACKAGES BOUND</td>
<td>67</td>
</tr>
<tr>
<td>DS NOUSE,NOCLOSE-HWM</td>
<td>101</td>
</tr>
<tr>
<td>AUTH.SUCC-PUBLIC</td>
<td>0</td>
</tr>
<tr>
<td>BIND PLAN (ADD)</td>
<td>0</td>
</tr>
<tr>
<td>BIND PACK (ADD)</td>
<td>0</td>
</tr>
<tr>
<td>DS NOUSE,NOCLOSE-CURR.</td>
<td>0</td>
</tr>
<tr>
<td>AUTOB.PLAN SUCC.</td>
<td>1</td>
</tr>
<tr>
<td>AUTOB.PACK SUCC.</td>
<td>0</td>
</tr>
<tr>
<td>DS NOUSE,NOCLOSE-CURR.</td>
<td>0</td>
</tr>
<tr>
<td>AUTOB.PLAN ATTMPT</td>
<td>1</td>
</tr>
<tr>
<td>AUTOB.PACK ATTMPT</td>
<td>0</td>
</tr>
<tr>
<td>R/N TO R/O CONVERSIONS</td>
<td>97</td>
</tr>
<tr>
<td>PKG-AUTH.SUCC-PUB</td>
<td>0</td>
</tr>
<tr>
<td>AUTOB.PLAN COMM</td>
<td>0</td>
</tr>
<tr>
<td>REBIND PACK COMM</td>
<td>0</td>
</tr>
<tr>
<td>PKG-AUTHID UNRTN</td>
<td>0</td>
</tr>
<tr>
<td>REBIND PLAN ATTMPT</td>
<td>0</td>
</tr>
<tr>
<td>REBIND PACK ATTMPT</td>
<td>0</td>
</tr>
<tr>
<td>PKG-ENTRY UNRTN</td>
<td>0</td>
</tr>
<tr>
<td>PLANS REBOUND</td>
<td>0</td>
</tr>
<tr>
<td>PACKAGES REBOUND</td>
<td>0</td>
</tr>
<tr>
<td>RTN-AUTH.SUCC</td>
<td>2</td>
</tr>
<tr>
<td>FREE PLAN COMMND</td>
<td>0</td>
</tr>
<tr>
<td>FREE PACKAGE COM</td>
<td>0</td>
</tr>
<tr>
<td>RTN-AUTH.SUCC-PUB</td>
<td>0</td>
</tr>
<tr>
<td>FREE PLAN ATTMPT</td>
<td>0</td>
</tr>
<tr>
<td>FREE PACK ATTMPT</td>
<td>0</td>
</tr>
<tr>
<td>RTN-AUTH.UNSUCC</td>
<td>6</td>
</tr>
<tr>
<td>PLANS FREED</td>
<td>0</td>
</tr>
<tr>
<td>PACKAGES FREED</td>
<td>0</td>
</tr>
<tr>
<td>RTN-AUTHID UNRTN</td>
<td>0</td>
</tr>
<tr>
<td>TEST BENDS</td>
<td>0</td>
</tr>
<tr>
<td>AUTOB.JVN,RES.ID</td>
<td>0</td>
</tr>
<tr>
<td>RTN-ENTRY UNRTN</td>
<td>0</td>
</tr>
<tr>
<td>QTREDPN</td>
<td>2346</td>
</tr>
</tbody>
</table>

**PLAN ALLOC.ATTMP**

The number of times a request was made to allocate a bound plan for an agent.

It represents the number of times DB2 was requested to create a thread by the attachment facility for the user. This does not include allocations for DB2 system agents.

**Field Name:** QTALLOCA

**PACK.ALLOC.ATTMP**

The number of attempts to allocate a package.

**Field Name:** QTPKALLA

**OPEN DATASETS - HWM**

The maximum number of data sets concurrently open since the last time DB2 was started. This is a high-water mark (HWM).

**Background and Tuning Information**

Monitor this field to see whether you are reaching the maximum number of open data sets permissible. The maximum number currently is 10000.

**Field Name:** QTMAXDS

This is an exception field.

**AUTHORIZ.ATTEMPTS**

The number of authorization checks performed for plans, packages, and stored procedures since DB2 was started. This includes successful and failed checks.

**Field Name:** QTAUCHK

This is an exception field.

**PLAN ALLOC.SUCC.**

The number of successful plan allocation attempts.
The cause of plan allocation failure could be plan unavailability or attempting to allocate a nonexistent plan.

**Field Name:** QTALLOC

**PACK.ALLOC.SUCC.**

The number of successful package allocation attempts.

**Background and Tuning Information**

Package allocation failure can occur when a package is unavailable or does not exist.

A high count of the number of packages unsuccessfully allocated (QTPKALLA - QTPKALL) typically occurs when a package list with multiple collections is used and frequently-used packages are found in the back end rather than in the front end of a package list. For example, when a package is found in the tenth collection, QTPKALLA is incremented by 10, one for each collection searched, but QTPKALL is incremented by 1.

A high number of packages unsuccessfully allocated can be accompanied by a high count of the number of unsuccessful checks for package execute authority made using the package authorization check because an application entry was not found in the cache (QTPACNOT). In this case, placing frequently used packages in the front end of a package list would reduce the number of Buffer Manager Getpages to the catalog/directory tables.

**Field Name:** QTPKALL

**OPEN DATASETS - CURR.**

The number of data sets concurrently open (snapshot).

**Field Name:** QTDSOPN

**AUTHORIZ.SUCCESS.**

The number of successful authorization checks performed on plans, packages, and stored procedures, since DB2 was started.

**Field Name:** QTAUSUC

This is an exception field.

**PLANS BOUND**

The number of plans successfully bound and kept for future agent allocations.

This field represents the sum of successful BIND ADD (QTBINDA) and successful BIND REPLACE (QTBINDR) commands. This counter is not incremented for BIND subcommands with no plan ID specified, as identified by QTTESTB. Note that QTBINDA + QTBINDR is not necessarily equal to this field. It is equal only if all BIND ADD and BIND REPLACE subcommands issued are successful.

**Field Name:** QTPLNBD

This is an exception field.

**PACKAGES BOUND**

The number of packages bound and kept for future package allocations.
It is the sum of successful BIND ADD PACKAGE and BIND REPLACE PACKAGE subcommands, but only if all these commands are really issued successfully.

Field Name: QTPKGBD

DS NOUSE,NOCLOSE-HWM

The maximum number of data sets on the deferred close queue. It is a high-water mark representing the maximum number of data sets that are not in use but have not been physically closed yet.

Field Name: QTMAXPB

This is an exception field.

AUTH.SUCC-NO CAT.

The number of successful authorization checks that do not use the DB2 catalog (including plan cache checks and public checks).

Background and Tuning Information

For transaction level security, ENABLE and DISABLE on BIND PACKAGE should be used to ensure adequate security. Granting execute authority on the plan to public should be adequate.

Field Name: QTAUCCH

BIND PLAN (ADD)

The number of successful and unsuccessful BIND ADD subcommands issued.

The sum of QTBINDA, QTBINDR, and QTTESTB equals the total number of BIND subcommands.

Field Name: QTBINDA

BIND PACK (ADD)

The number of successful and unsuccessful BIND ADD PACKAGE subcommands issued.

Field Name: QTBINDPA

DS NOUSE,NOCLOSE-CURR.

The number of data sets that are not currently used, but are not closed due to a deferred close (snapshot).

Field Name: QTSLWDD

This is an exception field.

AUTH.SUCC-PUBLIC

The number of successful authorization checks based on EXECUTE authority granted to PUBLIC.

Field Name: QTAUPUB

This is an exception field.

BIND PLAN (REPL)

The number of successful and unsuccessful BIND REPLACE subcommands issued.

Field Name: QTBINDR
IFCID 002 - Service Controller Data

This is an exception field.

BIND PACK (REPL)

The number of successful and unsuccessful BIND REPLACE PACKAGE subcommands issued.

Field Name: QTBINDPR

This is an exception field.

DS CLOSED-THRESH.REACH

The number of data sets that were closed because the total number of open data sets reached the deferred close threshold value. The deferred close value is based on the value of DSMAX or the MVS DD limit (whichever is smaller).

Field Name: QTDSDRN

This is an exception field.

PKG-AUTH.SUCC

The number of successful package EXECUTE authorization checks without accessing the DB2 catalog.

Field Name: QTPACAUT

AUTOB.PLAN ATTMP

The number of attempts to autobind a plan. This occurs when the plan was invalidated by modifications to the declarations of the data referenced by the programs bound as part of the plan. For example, dropping an index when it is used in the plan results in automatic bind.

Field Name: QTABINDA

AUTOB.PACK ATTMP

The number of attempts to autobind a package.

Background and Tuning Information

If YES was specified, or defaulted, for autobind on DB2 install panel DSNTIPB, an autobind occurs when a plan or package:

• Is invalid because declarations of the data referenced by the program or package were modified. For example, when an index used in a package is dropped, an automatic bind occurs when the package is run for the first time after the index was dropped.

• Was bound in a later release and is used in a previous release for the first time.

• Was used in a previous release but is later remigrated and used in a later release for the first time.

Field Name: QTAUTOBA

This is an exception field.

R/W TO R/O CONVERSIONS

The number of infrequently updated data sets that are converted from R/W to R/O state. An updated data set is considered infrequently updated when it has not been updated for either 5 consecutive DB2 checkpoints or 60 minutes. For tablespace data sets, the switching from R/W to R/O state means the SYSLGRNG entry is closed.
Field Name: QTPCCT
This is an exception field.

PKG-AUTH.SUCC-PUB
The number of successful package EXECUTE authorization checks without accessing the DB2 catalog. Package EXECUTE authority was granted to PUBLIC in the package authorization cache.

Field Name: QTPACPUB

AUTOB.PLAN SUCC.
The number of plans successfully autobound.

Field Name: QTABIND

AUTOB.PACK SUCC.
The number of packages successfully autobound.

Field Name: QTPKABND

PKG-AUTH.UNSUCC
The number of unsuccessful package EXECUTE authorization checks in the package authorization cache. No applicable entry was found in the cache and DB2 catalog access was used.

Background and Tuning Information
This field indicates the efficiency of the cache. It should be close to zero.

Field Name: QTPACNOT

REBIND PLAN COMM
The number of REBIND subcommands issued. More than one plan can be rebound with a single REBIND subcommand. If the value in this field is 1, the number of plans you are attempting to rebind is shown in the Rebind - plan attempts field.

Field Name: QTREBIND

REBIND PACK COMM
The number of REBIND PACKAGE subcommands issued. More than one package can be rebound with a single subcommand. If the value in this field is 1, Rebind - package attempts shows the number of packages you are attempting to rebind.

Field Name: QTRBINDP

PKG-AUTHID OWRTN
The number of times an authorization ID was overwritten to add another one to the package authorization cache.

Field Name: QTPACOW1

REBIND PLAN ATTM
The number of attempts to rebind a plan. This number can be larger than the value shown in the Rebind - plan subcommands field because you can specify more than one plan in a single REBIND subcommand.

Field Name: QTRBINDA

REBIND PACK ATTM
The number of attempts to rebind a package. This can be larger than the value shown in Rebind package subcommands because you can rebind more than one package with a single command.

Field Name: QTRBNDPA

PKG-ENTRY OWRTN

The number of times an entry for a collection-ID or package-ID was overwritten to add another one to the package authorization cache.

Field Name: QTPACOW2

PLANS REBOUND

The number of rebind attempts that completed successfully. This field is equal to the Rebind - Plan attempts field if all specified plans rebound successfully.

Field Name: QTPLNRBD

PACKAGES REBOUND

The number of packages successfully rebound. If all specified packages were rebound successfully, this field is equal to Rebind package attempts.

Field Name: QTPKGRBD

RTN-AUTH.SUCC

The number of times the routine authorization cache was checked successfully of EXECUTE authority on a stored procedure or user-defined function. The DB2 catalog was not accessed. This counter includes the number of PUBLIC authorization checks.

Field Name: QTRACAUT

FREE PLAN COMMND

The number of FREE subcommands issued.

More than one plan can be freed with a single FREE subcommand. If this field is 1, then the number of plans you are trying to free is shown in ATTEMPTS TO FREE A PLAN.

Field Name: QTFREE

FREE PACKAGE COM

The number of FREE PACKAGE subcommands issued.

More than one package can be freed with a single FREE subcommand. If the value in this field is 1, then the number of packages you are attempting to free is shown in ATTEMPTS TO FREE A PACKAGE.

Field Name: QTFREEP

RTN-AUTH.SUCC-PUB

Number of successful authorization checks for user-defined function or stored procedure execution authority when that authority is held by PUBLIC. The DB2 catalog was not checked.

Field Name: QTRACPUB

FREE PLAN ATTMPT

The number of attempts to free a plan.
This value can be larger than FREE PLAN SUBCOMMANDS because multiple plan IDs can be specified in a single FREE subcommand.

**Field Name:** `QTFREEA`

**FREE PACK ATTMPT**

The number of attempts to free a package. This number can be larger than FREE PACKAGE SUBCOMMANDS because you can free several packages with a single command.

**Field Name:** `QTFREEAP`

**RTN-AUTH.UNSUCC**

Number of unsuccessful authorization checks for user-defined function or stored procedure EXECUTE authority because no applicable entry was found in the routine authorization cache.

**Background and Tuning Information**

This field indicates the efficiency of the cache. It should be close to zero.

**Field Name:** `QTRACNOT`

**PLANS FREED**

The number of times a plan was successfully freed.

Freeing a plan can fail if someone else is using the plan and holds a lock on it.

**Field Name:** `QTPLNFRD`

**PACKAGES FREED**

The number of times a package was successfully freed. If all the specified packages were freed successfully, the value of this field is equal to ATTEMPTS TO FREE A PACKAGE.

**Field Name:** `QTPKGFRD`

**RTN-AUTHID OWRTN**

Number of times an individual authorization ID was overwritten in an entry of the routine authorization cache.

**Field Name:** `QTACOW1`

**TEST BINDS**

The number of BIND subcommands issued without a plan ID.

**Field Name:** `QTTESTB`

**AUTOB.INV.RES.ID**

The number of requests to allocate a nonexistent plan or package. This is the number of all plan and package allocation attempts that failed because the resource was unavailable or the object did not exist.

**Field Name:** `QTINVRID`

**RTN-ENTRY OWRTN**

Number of times that DB2 overwrote a routine entry in the routine authorization cache.

An entry in the routine authorization cache can refer to a function or procedure or to all functions or procedures within a specific schema.
IFCID 002 - Service Controller Data

Field Name: QTRACOW2
RTN-CACHE NO ADD

Number of times that DB2 could not add an entry to the routine authorization cache.

An entry in the routine authorization cache can refer to a function or procedure or to all functions or procedures within a specific schema.

Field Name: QTRACNAC
IFCID 002 - SQL Call Data

This topic shows detailed information about “Record Trace - IFCID 002 - SQL Call Data”.

Record trace - IFCID 002 - SQL Call Data

The field labels shown in the following sample layout of “Record Trace - IFCID 002 - SQL Call Data” are described in the following section.

<table>
<thead>
<tr>
<th>Field</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECT</td>
<td>12158358</td>
</tr>
<tr>
<td>INSERT</td>
<td>2317413</td>
</tr>
<tr>
<td>UPDATE</td>
<td>30540</td>
</tr>
<tr>
<td>OPEN</td>
<td>138162</td>
</tr>
<tr>
<td>CLOSE</td>
<td>126456</td>
</tr>
<tr>
<td>LOCK TABLE</td>
<td>0</td>
</tr>
<tr>
<td>FREE LOCATOR</td>
<td>0</td>
</tr>
<tr>
<td>GRANT</td>
<td>0</td>
</tr>
<tr>
<td>UNLOCK TABLE</td>
<td>0</td>
</tr>
<tr>
<td>ALLOCATE CURSOR</td>
<td>5</td>
</tr>
<tr>
<td>REVOKE</td>
<td>0</td>
</tr>
<tr>
<td>ASSOCIATE LOCATOR</td>
<td>5</td>
</tr>
<tr>
<td>DECRIBED</td>
<td>972</td>
</tr>
<tr>
<td>RELEASE</td>
<td>0</td>
</tr>
<tr>
<td>PREPARE</td>
<td>30540</td>
</tr>
<tr>
<td>OPEN</td>
<td>138162</td>
</tr>
<tr>
<td>CLOSE</td>
<td>126456</td>
</tr>
<tr>
<td>FETCH</td>
<td>7413786</td>
</tr>
<tr>
<td>DESCRIBE TABLE</td>
<td>0</td>
</tr>
<tr>
<td>CONNECT TYPE 1</td>
<td>0</td>
</tr>
<tr>
<td>CONNECT TYPE 2</td>
<td>15</td>
</tr>
<tr>
<td>CREATE DATABASE</td>
<td>0</td>
</tr>
<tr>
<td>DROP DATABASE</td>
<td>0</td>
</tr>
<tr>
<td>CREATE STOGROUP</td>
<td>0</td>
</tr>
<tr>
<td>DROP STOGROUP</td>
<td>0</td>
</tr>
<tr>
<td>CREATE TABSPACE</td>
<td>0</td>
</tr>
<tr>
<td>DROP TABSPACE</td>
<td>0</td>
</tr>
<tr>
<td>CREATE INDEX</td>
<td>6</td>
</tr>
<tr>
<td>DROP INDEX</td>
<td>0</td>
</tr>
<tr>
<td>CREATE VIEW</td>
<td>12</td>
</tr>
<tr>
<td>DROP VIEW</td>
<td>0</td>
</tr>
<tr>
<td>CREATE SEQUENCE</td>
<td>6</td>
</tr>
<tr>
<td>DROP SEQUENCE</td>
<td>0</td>
</tr>
<tr>
<td>CREATE TRIGGER</td>
<td>6</td>
</tr>
<tr>
<td>DROP TRIGGER</td>
<td>0</td>
</tr>
<tr>
<td>CREATE IDENT</td>
<td>6</td>
</tr>
<tr>
<td>DROP IDENT</td>
<td>0</td>
</tr>
<tr>
<td>CREATE FUNCTION</td>
<td>6</td>
</tr>
<tr>
<td>DROP FUNCTION</td>
<td>0</td>
</tr>
<tr>
<td>CREATE PROCEDURE</td>
<td>6</td>
</tr>
<tr>
<td>DROP PROCEDURE</td>
<td>0</td>
</tr>
<tr>
<td>CREATE ROLE</td>
<td>6</td>
</tr>
<tr>
<td>DROP ROLE</td>
<td>0</td>
</tr>
<tr>
<td>CREATE TRUST CONT</td>
<td>6</td>
</tr>
<tr>
<td>DROP TRUST CONT</td>
<td>0</td>
</tr>
<tr>
<td>CREATE MASK/PERM</td>
<td>0</td>
</tr>
<tr>
<td>DROP MASK/PERM</td>
<td>0</td>
</tr>
<tr>
<td>CREATE VARIABLE</td>
<td>0</td>
</tr>
<tr>
<td>DROP VARIABLE</td>
<td>0</td>
</tr>
<tr>
<td>CREATE PACKAGE</td>
<td>0</td>
</tr>
<tr>
<td>DROP PACKAGE</td>
<td>0</td>
</tr>
<tr>
<td>SET CUR SQL ID</td>
<td>25</td>
</tr>
<tr>
<td>SET CUR PATH</td>
<td>0</td>
</tr>
<tr>
<td>SET CUR VAR</td>
<td>0</td>
</tr>
<tr>
<td>SET CUR VARS</td>
<td>0</td>
</tr>
<tr>
<td>SET CUR RULES</td>
<td>0</td>
</tr>
<tr>
<td>SET CUR PRECISION</td>
<td>0</td>
</tr>
</tbody>
</table>

**SELECT**

The number of SQL SELECT statements executed.

**Field Name:** QXSELECT

**INSERT**

The number of INSERT statements executed.

**Field Name:** QXINSRT

**UPDATE**

The number of UPDATE statements executed.

**Field Name:** QXUPDTE

**DELETE**

The number of DELETE statements executed.

**Field Name:** QXDELET

**DESCRIPTION**

MULTI-ROW PROCESSING:

- ROWS FETCHED: 7596806
- ROWS INSERTED: 2915959
- ROWS UPDATED: 3080851
- ROWS DELETED: 406827

Chapter 40. IFCID Record Blocks  40-135
IFCID 002 - SQL Call Data

The number of DESCRIBE, DESCRIBE CURSOR, DESCRIBE INPUT, and DESCRIBE PROCEDURE statements executed. This number at the server location might not match the user application because of DDF's internal processing.

Field Name: QXDESC

PREPARE

The number of SQL PREPARE statements executed. This number at the server location might not match the user application because of DDF's internal processing.

Field Name: QXPREP

OPEN

The number of OPEN statements executed.

Field Name: QXOPEN

CLOSE

The number of CLOSE statements executed. This number at the server location might not match the user application because of DDF's internal processing.

Field Name: QXCLOSE

FETCH

The number of FETCH statements executed. This number at the server location might not match the user application because of DDF's internal processing.

Field Name: QXFETCH

COMMENT ON

The number of COMMENT ON statements executed.

Field Name: QXCMTON

LOCK TABLE

The number of LOCK TABLE statements executed.

Field Name: QXLOCK

GRANT

The number of GRANT statements executed.

Field Name: QXGRANT

REVOKE

The number of REVOKE statements executed.

Field Name: QXREVOK

INCREMENTAL Binds

The number of incremental binds (excluding prepare). It is incremented by:

- SQL statements with BIND VALIDATE(RUN) that fail at bind time and are bound again at execution time
- Static DDL statements (such as CREATE TABLE, DROP TABLE, LOCK TABLE) that use DB2 private protocol
Background and Tuning Information

If a plan is bound with VALIDATE(RUN), DB2 performs validity checks at bind time and rechecks any failures at run time. This can result in catalog contention and degraded application performance, depending on the number of statements flagged and how many times they are executed. Avoid VALIDATE(RUN) if possible. Ensure that all objects are created and all privileges are granted before bind, and select the VALIDATE(BIND) option.

Field Name: QXINCRB
This is an exception field.

LABEL ON
The number of LABEL ON statements executed.

Field Name: QXLABON
This is an exception field.

DESCRIBE TABLE
The number of DESCRIBE TABLE statements executed.

Field Name: QXDSCRTB
This is an exception field.

CONNECT TYPE 1
The number of CONNECT type 1 statements executed.

Field Name: QXCON1
This is an exception field.

CONNECT TYPE 2
The number of CONNECT type 2 statements executed.

Field Name: QXCON2
This is an exception field.

RELEASE
The number of RELEASE statements executed.

Field Name: QXREL
This is an exception field.

ASSOCIATE LOCATOR
The number of SQL ASSOCIATE LOCATORS statements executed.

Field Name: QXALOCL
This is an exception field.

ALLOCATE CURSOR
The number of SQL ALLOCATE CURSOR statements executed.

Field Name: QXALOCC
This is an exception field.

RENAME TABLE
IFCID 002 - SQL Call Data

The number of RENAME TABLE statements executed.
Field Name: QXRNTAB
This is an exception field.

HOLD LOCATOR
The number of SQL HOLD LOCATOR statements executed.
Field Name: QXHOLDL

FREE LOCATOR
The number of SQL FREE LOCATOR statements executed.
Field Name: QXFREEL

MERGE
The number of times a MERGE statement was executed.
Field Name: QXMERGE

TRUNCATE TABLE
The number of TRUNCATE TABLE statements issued.
Field Name: QXTRTBL

RENAME INDEX
The number of RENAME INDEX statements issued.
Field Name: QXRNIX

CREATE DATABASE
The number of CREATE DATABASE statements executed.
Field Name: QXCRDAB

DROP DATABASE
The number of DROP DATABASE statements executed.
Field Name: QXDRPDB

ALTER DATABASE
The number of ALTER DATABASE statements executed.
Field Name: QXALDAB

CREATE STOGROUP
The number of CREATE STOGROUP statements executed.
Field Name: QXCRSTG

DROP STOGROUP
The number of DROP STOGROUP statements executed.
Field Name: QXDRPST

ALTER STOGROUP
The number of ALTER STOGROUP statements executed.
Field Name: QXALTST

CREATE TABSPACE
IFCID 002 - SQL Call Data

The number of CREATE TABLESPACE statements executed.
Field Name: QXCTABS

DROP TABSPACE
The number of DROP TABLESPACE statements executed.
Field Name: QXDRPTS

ALTER TABSPACE
The number of ALTER TABLESPACE statements executed.
Field Name: QXALTTS

CREATE TABLE
The number of CREATE TABLE statements executed.
Field Name: QXCRTAB

DROP TABLE
The number of DROP TABLE statements executed.
Field Name: QXDRPTA

ALTER TABLE
The number of ALTER TABLE statements executed.
Field Name: QXALTTA

CREATE AUX TABLE
The number of CREATE AUXILIARY TABLE statements executed.
Field Name: QXCRATB

CREATE TMP TABLE
The number of CREATE GLOBAL TEMPORARY TABLE statements executed.
Field Name: QXCRGTT

DECLARE TMP TABLE
The number of DECLARE GLOBAL TEMPORARY TABLE statements executed.
Field Name: QXDCLGTT

CREATE INDEX
The number of CREATE INDEX statements executed.
Field Name: QXCRINX

DROP INDEX
The number of DROP INDEX statements executed.
Field Name: QXDRPIX

ALTER INDEX
The number of ALTER INDEX statements executed.
Field Name: QXALTIX

CREATE VIEW
The number of CREATE VIEW statements executed.
**Field Name:** QXDEFVU

**DROP VIEW**
The number of DROP VIEW statements executed.
**Field Name:** QXDRPVU

**ALTER VIEW**
The number of ALTER VIEW statements issued.
**Field Name:** QXALTVW

**CREATE SYNONYM**
The number of CREATE SYNONYM statements executed.
**Field Name:** QXCRSYN

**DROP SYNONYM**
The number of DROP SYNONYM statements executed.
**Field Name:** QXDRPSY

**CREATE ALIAS**
The number of CREATE ALIAS statements executed.
**Field Name:** QXCRALS

**DROP ALIAS**
The number of SQL DROP ALIAS statements executed.
**Field Name:** QXDRPAL

**CREATE SEQUENCE**
The number of CREATE SEQUENCE statements.
**Field Name:** QXCRESEQ

**DROP SEQUENCE**
The number of ALTER SEQUENCE statements.
**Field Name:** QXALTSEQ

**ALTER SEQUENCE**
The number of DROP SEQUENCE statements.
**Field Name:** QXDROSEQ

**CREATE TRIGGER**
The number of SQL CREATE TRIGGER statements.
**Field Name:** QXCRTRIG

**DROP TRIGGER**
The number of DROP TRIGGER statements executed.
**Field Name:** QXDRPTR

**CREATE DIST TYPE**
The number of CREATE DISTINCT TYPE statements executed.
**IFCID 002 - SQL Call Data**

Field Name: QXCDIST

**DROP DIST TYPE**
The number of DROP DISTINCT TYPE statements executed.

Field Name: QXDDIST

**CREATE FUNCTION**
The number of CREATE FUNCTION statements executed.

Field Name: QXCRUDF

**DROP FUNCTION**
The number of DROP FUNCTION statements executed.

Field Name: QXDRPFN

**ALTER FUNCTION**
The number of DROP DISTINCT TYPE statements executed.

Field Name: QXDDIST

**CREATE PROCEDURE**
The number of CREATE PROCEDURE statements issued.

Field Name: QXCRPRO

**DROP PROCEDURE**
The number of DROP PROCEDURE statements executed.

Field Name: QXDRPPR

**ALTER PROCEDURE**
The number of DROP DISTINCT TYPE statements executed.

Field Name: QXDDIST

**CREATE ROLE**
The number of CREATE ROLE statements executed.

Field Name: QXCRROL

**DROP ROLE**
The number of DROP ROLE statements issued.

Field Name: QXDRPROL

**CREATE TRUST CONT**
The number of CREATE TRUSTED CONTEXT statements issued.

Field Name: QXCRCTX

**DROP TRUST CONT**
The number of DROP TRUSTED CONTEXT statements issued.

Field Name: QXDRPCTX

**ALTER TRUST CONT**
The number of ALTER TRUSTED CONTEXT statements issued.

Field Name: QXALTCTX
CREATE MASK/PERM
The number of CREATE MASK and CREATE PERMISSION statements executed.
Field Name: QXCREMP

DROP MASK/PERM
The number of DROP MASK and DROP PERMISSION statements executed.
Field Name: QXDRPMP

ALTER MASK/PERM
The number of ALTER MASK and ALTER PERMISSION statements executed.
Field Name: QXALTMP

CREATE VARIABLE
The number of CREATE VARIABLE statements.
Field Name: QXCRTSV

DROP VARIABLE
The number of DROP VARIABLE statements.
Field Name: QXDRPSV

DROP PACKAGE
The number of SQL DROP PACKAGE statements executed.
Field Name: QXDRPPKG

ALTER JAR
The number of ALTER JAR statements issued.
Field Name: QXALTJR

SET CUR SQL ID
The number of SET CURRENT SQLID statements executed.
Field Name: QXSETPSQL

SET HOST VAR
The number of SET HOST VARIABLE statements executed. The special register that was retrieved is not tracked.
Field Name: QXSETHV

SET CONNECTION
The number of SET CONNECTION statements executed.
Field Name: QXSETCON

SET CUR DEGREE
The number of SET CURRENT DEGREE statements executed.
Field Name: QXSETCDG

SET CUR RULES
The number of SET CURRENT RULES statements executed.
Field Name: QXSETCRL

SET CUR PATH
The number of SET CURRENT PATH statements executed.

Field Name: QXSETPTH

SET CUR PRECISION
The number of SET CURRENT PRECISION statements executed.

Field Name: QXSETCPR

ROWS FETCHED
The number of rows fetched.

Field Name: QXRWSFETCHD

ROWS INSERTED
The number of rows inserted.

Field Name: QXRWSINSRTD

ROWS UPDATED
The number of rows updated.

Field Name: QXRWSUPDTD

ROWS DELETED
The number of rows deleted.

Field Name: QXRWSDELETED
Accounting shows the data from IFCID 003. 

- "IFCID 003 - Accelerator Data" on page 40-146
  This topic shows detailed information about "Record Trace - IFCID 003 - Accelerator Data".

- "IFCID 003 - Buffer Manager Accounting Data" on page 40-149
  This topic shows detailed information about "Record Trace - IFCID 003 - Buffer Manager Accounting Data".

- "IFCID 003 - Data Sharing Accounting Data" on page 40-153
  This topic shows detailed information about "Record Trace - IFCID 003 - Data Sharing Accounting Data".

- "IFCID 003 - Data Sharing Locking Data" on page 40-154
  This topic shows detailed information about "Record Trace - IFCID 003 - Data Sharing Locking Data".

- "IFCID 003 - DDF Data by Location" on page 40-156
  This topic shows detailed information about "Record Trace - IFCID 003 - DDF Data by Location".

- "IFCID 003 - Dynamic SQL Statement" on page 40-165
  This topic shows detailed information about "Record Trace - IFCID 003 - Dynamic SQL Statement".

- "IFCID 003 - Group Buffer Pools Activity Data" on page 40-167
  This topic shows detailed information about "Record Trace - IFCID 003 - Group Buffer Pools Activity Data".

- "IFCID 003 - IFI Class 5 Times and Data Capture" on page 40-169
  This topic shows detailed information about "Record Trace - IFCID 003 - IFI Class 5 Times and Data Capture".

- "IFCID 003 - Initial DB2 Requester and MVS Correlation Data" on page 40-171
  This topic shows detailed information about "Record Trace - IFCID 003 - Initial DB2 Requester and MVS Correlation Data".

- "IFCID 003 - Instrumentation Accounting Data" on page 40-175
  This topic shows detailed information about "Record Trace - IFCID 003 - Instrumentation Accounting Data".

- "IFCID 003 - Instrumentation Accounting Data Overflow" on page 40-190
  This topic shows detailed information about "Record Trace - IFCID 003 - Instrumentation Accounting Data Overflow".

- "IFCID 003 - Locking Data" on page 40-193
  This topic shows detailed information about "Record Trace - IFCID 003 - Locking Data".

- "IFCID 003 - Logging" on page 40-197
  This topic shows detailed information about "Record Trace - IFCID 003 - Logging".

- "IFCID 003 - Miscellaneous" on page 40-198
  This topic shows detailed information about "Record Trace - IFCID 003 - Miscellaneous".

- "IFCID 003 - Nested SQL Activity" on page 40-199
  This topic shows detailed information about "Record Trace - IFCID 003 - Nested SQL Activity".

- "IFCID 003 - Query Parallelism" on page 40-201
  This topic shows detailed information about "Record Trace - IFCID 003 - Query Parallelism".
This topic shows detailed information about “Record Trace - IFCID 003 - Resource Limit Facility”.

This topic shows detailed information about “Record Trace - IFCID 003 - RID List Processing”.

This topic shows detailed information about “Record Trace - IFCID 003 - Rollup Accounting Correlation Block (DB2 10 or later)”.

This topic shows detailed information about “Record Trace - IFCID 003 - ROWID”.

This topic shows detailed information about “Record Trace - IFCID 003 - SQL Call Data”.

IFCID 003 - Accounting
IFCID 003 - Accelerator Data

IFCID 003 - Accelerator Data

This topic shows detailed information about “Record Trace - IFCID 003 - Accelerator Data”.

Record trace - IFCID 003 - Accelerator Data

The field labels shown in the following sample layout of “Record Trace - IFCID 003 - Accelerator Data” are described in the following section.

**SERVER ID**

The accelerator server identifier.

Field Name: Q8ACNAME

**PRODUCT ID**

The accelerator product identifier.

Field Name: Q8ACPRID

**CONNECTS TO ACCELERATOR**

The number of accelerator connects.

Field Name: Q8ACCONN

**REQUESTS SENT TO ACCELERATOR**

The number of accelerator requests.

Field Name: Q8ACREQ

**REQUESTS SENT TO ACCELERATOR - TIMED OUT**

The number of timed out requests.

Field Name: Q8ACTOUT

**REQUESTS SENT TO ACCELERATOR - FAILED**

The number of failed requests.

Field Name: Q8ACFAIL

**BYTES SENT TO ACCELERATOR**

The number of bytes sent.

Field Name: Q8ACBYTES

**MESSAGES SENT TO ACCELERATOR**

The number of messages sent.

Field Name: Q8ACMSGS

**BLOCKS SENT TO ACCELERATOR**

The number of blocks sent.

Field Name: Q8ACBYTS

**BYTES RECEIVED FROM ACCELERATOR**

The number of bytes received.

Field Name: Q8ACBYTS

**MESSAGES RECEIVED FROM ACCELERATOR**

The number of messages received.

Field Name: Q8ACMSGS

**BLOCKS RECEIVED FROM ACCELERATOR**

The number of blocks received.

Field Name: Q8ACBYTS

**SERVER ID**

The accelerator server identifier.

Field Name: Q8ACNAME

**PRODUCT ID**

The accelerator product identifier.

Field Name: Q8ACPRID

**CONNECTS TO ACCELERATOR**

The number of accelerator connects.

Field Name: Q8ACCONN

**REQUESTS SENT TO ACCELERATOR**

The number of accelerator requests.

Field Name: Q8ACREQ

**REQUESTS SENT TO ACCELERATOR - TIMED OUT**

The number of timed out requests.

Field Name: Q8ACTOUT

**REQUESTS SENT TO ACCELERATOR - FAILED**

The number of failed requests.

Field Name: Q8ACFAIL

**BYTES SENT TO ACCELERATOR**

The number of bytes sent.

Field Name: Q8ACBYTES

**MESSAGES SENT TO ACCELERATOR**

The number of messages sent.

Field Name: Q8ACMSGS

**BLOCKS SENT TO ACCELERATOR**

The number of blocks sent.

Field Name: Q8ACBYTS

**BYTES RECEIVED FROM ACCELERATOR**

The number of bytes received.

Field Name: Q8ACBYTS

**MESSAGES RECEIVED FROM ACCELERATOR**

The number of messages received.

Field Name: Q8ACMSGS

**BLOCKS RECEIVED FROM ACCELERATOR**

The number of blocks received.

Field Name: Q8ACBYTS

**ACCELERATOR SVCS TCP/IP CPU TIME**

The accumulated time spent on CPU.

Field Name: Q8ACCPU

**ACCELERATOR SVCS TCP/IP ELAPSED TIME**

The elapsed time for the ACCELERATOR.

Field Name: Q8ACElapsedTime

**ACCUMULATED ACCELERATOR CPU TIME**

The accumulated time spent on CPU.

Field Name: Q8ACCPU

**ACCUMULATED ACCELERATOR ELAPSED TIME**

The elapsed time for the ACCELERATOR.

Field Name: Q8ACElapsedTime

**ACCUMULATED ACCELERATOR WAIT TIME**

The accumulated time spent on wait.

Field Name: Q8ACWaitTime
IFCID 003 - Accelerator Data

The number of blocks sent.
Field Name: Q8ACBLKS

ROWS SENT TO ACCELERATOR
The number of rows sent.
Field Name: Q8ACROWS

BYTES RECEIVED FROM ACCELERATOR
The number of bytes returned.
Field Name: Q8ACBYTR

MESSAGES RECEIVED FROM ACCELERATOR
The number of messages returned.
Field Name: Q8ACMSGR

BLOCKS RECEIVED FROM ACCELERATOR
The number of blocks returned.
Field Name: Q8ACBLKR

ROWS RECEIVED FROM ACCELERATOR
The number of rows returned.
Field Name: Q8ACROWR

ACCELERATOR SVCS TCPIP CPU TIME
The accelerator services TCP/IP CPU time measured in DB2 for the
amount of CPU consumed by the DDF service task to perform the SEND
and RECEIVE to an accelerator service. It does not account for the TCP/IP
address CPU to route the message on to the network and receive the reply
into the DDF task.
Field Name: Q8ACTCPU

ACCUMULATED ACCELERATOR CPU TIME
The CPU time spent in the accelerator when executing requests from the
DB2 subsystem.
Field Name: Q8ACACPU

ACCUMULATED ACCELERATOR WAIT TIME
The wait time spent in the accelerator when executing requests from the
DB2 subsystem.
Field Name: Q8ACAWAT

ACCELERATOR SVCS TCPIP ELAPSED TIME
The accelerator services TCP/IP elapsed time measured in DB2. It starts
when sending the requests to the accelerator and ends when receiving the
results from the accelerator.
Field Name: Q8ACTELA

ACCUMULATED ACCELERATOR ELAPSED TIME
The elapsed time spent in the accelerator when executing requests from the
DB2 subsystem.
IFCID 003 - Accelerator Data

Field Name: Q8ACAELA
IFCID 003 - Buffer Manager Accounting Data

This topic shows detailed information about “Record Trace - IFCID 003 - Buffer Manager Accounting Data”.

Record trace - IFCID 003 - Buffer Manager Accounting Data

The field labels shown in the following sample layout of “Record Trace - IFCID 003 - Buffer Manager Accounting Data” are described in the following section.

### BUFFER MANAGER ACCOUNTING DATA

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>BUFFER POOL ID</td>
<td>0</td>
</tr>
<tr>
<td>SYNCHRON. READ</td>
<td>0</td>
</tr>
<tr>
<td>GETPAGES</td>
<td>10</td>
</tr>
<tr>
<td>GETPAGES FAILED</td>
<td>0</td>
</tr>
<tr>
<td>BUFFER UPDATES</td>
<td>0</td>
</tr>
<tr>
<td>SYNCHRON. WRITE</td>
<td>0</td>
</tr>
</tbody>
</table>

### BUFFER POOL ID

The buffer pool ID used by this thread.

**Field Name:** QBACPID

### SYNCHRON. READ

The number of synchronous read I/O operations. DB2 increments this counter for each media manager synchronous physical read. Asynchronous I/O requests are not counted.

**Field Name:** QBACRIO

### GETPAGES

The number of Getpage requests. This counter is incremented by successful Getpage requests for queries processed in parallel for each thread and for all successful and unsuccessful Getpage requests for queries that are not processed in parallel.

**Background and Tuning Information**

Reducing the number of Getpages can improve DB2 performance by reducing the number of synchronous page reads. With fewer Getpages, the requested page is more likely to be returned from the buffer pool. CPU usage is also reduced.

Check the ratio of Getpages to SQL DML statements, as a rule of thumb, try and keep this ratio below six.

You might need to modify the database and query design, for example:

- Add indexes to tables to reduce the number of pages scanned.
- Reassess the number of tables used and denormalize them, if necessary.
  
  As an example, a large table with many columns can result in several pages being fetched to satisfy a simple query requesting just a few columns. Splitting such a table into several tables with fewer columns, tailored to queries, will result in fewer pages returned for each query.
- Use correlated rather than noncorrelated queries to force the use of an index.

**Field Name:** QBACGET

This is an exception field.

### SEQ. PREFETCH

Chapter 40. IFCID Record Blocks 40-149
The number of SEQUENTIAL PREFETCH requests. This is incremented for each PREFETCH request. Each request can result in an I/O read. If it does, up to 32 pages can be read for SQL and up to 64 pages for utilities. For SQL, depending on the buffer pool size, a request does not result in an I/O if all the requested pages are already in the buffer pool.

DB2 can use sequential prefetch if the data is accessed in sequential order even though sequential prefetch was not requested at bind time. This is known as sequential detection and is not included in the sequential prefetch count. Sequential detection is included in dynamic prefetch requests field.

**Background and Tuning Information**

Table space scans and nonmatching index scans generally use sequential prefetch.

**Field Name:** QBACSEQ

This is an *exception* field.

**GETPAGES FAILED**

The number of times that a page requested for a query processed in parallel was unavailable because an I/O was in progress or the page was not found in the buffer pool. The agent does not wait, but control returns to the agent.

This counter is used only when queries are processed in parallel.

**Background and Tuning Information**

If this value is close to zero, most pages are already in the buffer pool, and wait time for synchronous I/O is small.

This counter can be high when, for example, there is a cluster index scan and the data is not truly clustered by the index key. In this instance, data pages are not accessed in their true order and the cluster ratio is not valid. Use the Runstats utility to update it.

The value of this field is also used to determine how many sequential prefetches of one page were scheduled.

**Field Name:** QBACNGT

**LIST PREFETCH**

The number of LIST PREFETCH requests.

**Special Considerations:**

1. List prefetch allows DB2 to access data pages efficiently even if the needed data pages are not contiguous. It can be used with single index access and is always used with multiple index access.
2. List prefetch is always used to access data from the inner table during a hybrid join.
3. Data pages are read in quantities equal to the sequential prefetch quantity, which depends on the buffer pool size and is usually 32 pages.
4. During bind time DB2 does not use list prefetch if the estimated number of RIDs to be processed would take more than 50% of the RID pool. During execution time, list prefetch processing terminates if DB2 detects that more than 25% of the rows in the table need to be accessed. If list prefetch is terminated, it is indicated in IFCID 125.
Field Name: QBACLPF
This is an exception field.

BUFFER UPDATES
The number of times a buffer update occurs. This is incremented every time a page is updated and is ready to be written to DASD. If the same page is updated twice, for example, the number is incremented by 2.

This number is kept for all types of pages including data pages and work-file pages.

Background and Tuning Information
A nonzero value indicates any of the following activities:
• SQL INSERT, UPDATE, or DELETE
• Merge scan join
• Internal sort activity on the work files

Check the access path to determine whether sort activity can be minimized or avoided.

Field Name: QBACSWS
This is an exception field.

DYNAMIC PREFETCH
The number of (dynamic) PREFETCH requests. This is triggered by sequential detection. This includes prefetches for segmented table spaces.

Background and Tuning Information
Dynamic prefetch is typically used for a SELECT or UPDATE that is run repeatedly, accessing the index for each access.

If sequential prefetch, list prefetch, and dynamic prefetch reads have large values, check whether the access path can be improved.

Field Name: QBACDPF
This is an exception field.

SYNCHRON. WRITE
The number of immediate (synchronous) write I/O operations.

Background and Tuning Information
Although an immediate write is rare, a small nonzero value is acceptable. A large value indicates that the system needs tuning.

Field Name: QBACIMW
This is an exception field.

PAGES READ ASYN-PAR
The number of asynchronous pages read by prefetch that the agent triggered.

Background and Tuning Information
This is used to determine the buffer pool hit ratio: (Getpage requests - Synchronous reads - Asynchronous pages read) / Getpage requests.

Field Name: QBACSIO
This is an exception field.
IFCID 003 - Data Sharing Accounting Data

This topic shows detailed information about “Record Trace - IFCID 003 - Data Sharing Accounting Data”.

Record trace - IFCID 003 - Data Sharing Accounting Data

The field labels shown in the following sample layout of “Record Trace - IFCID 003 - Data Sharing Accounting Data” are described in the following section.

DATA SHARING ACCOUNTING DATA
MEMBER NAMES: N/P

MEMBER NAMES

For an assisting task, the name of the parallelism coordinator. For a coordinating task, the name of each assisting member.

Field Name: QWDAXCQO
IFCID 003 - Data Sharing Locking Data

IFCID 003 - Data Sharing Locking Data
This topic shows detailed information about “Record Trace - IFCID 003 - Data Sharing Locking Data”.

Record trace - IFCID 003 - Data Sharing Locking Data
The field labels shown in the following sample layout of “Record Trace - IFCID 003 - Data Sharing Locking Data” are described in the following section.

<table>
<thead>
<tr>
<th>DATA SHARING LOCKING DATA</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>LOCK REQUESTS</td>
<td>5</td>
<td>10</td>
</tr>
<tr>
<td>UNSUSPENSIONS - IRLM</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>CHANGE REQUESTS</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>NOTIFY SENT</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

LOCK REQUESTS
The number of lock requests for P-locks.
Field Name: QTGALPLK

LOCK - XES
The number of P/L-lock requests propagated to z/OS XES synchronously.
This number is not incremented if the request is suspended before going to XES.
Field Name: QTGALSLM

SUSPENSIONS - IRLM
The number of suspensions due to IRLM global resource contention (IRLM lock states were in conflict).
Field Name: QTGAIGLO

UNLOCK REQUESTS
The number of unlock requests for P-locks.
Field Name: QTGAUPLK

UNLOCK - XES
The number of unlock requests propagated to z/OS XES.
Field Name: QTGAUSLM

SUSPENSIONS - XES
The number of suspensions due to z/OS XES global resource contention (z/OS XES lock states were in conflict whereas IRLM lock states were not).
Field Name: QTGASGLO

CHANGE REQUESTS
The number of change requests for P-locks.
Field Name: QTGACPLK

CHANGE - XES
The number of change requests propagated to z/OS XES.
Field Name: QTGACSLM

INCOMPATIBLE LOCKS
The number of global lock or change requests denied or suspended due to an incompatible retained lock.

Field Name: QTGADRTA

NOTIFY SENT
The number of notify messages sent.
Field Name: QTGANTFY

SYNC-ASYNC XES CONV
The total number of sync-to-async heuristic conversions for LOCK requests in XES. This conversion is done when XES determines that it is more efficient to drive the request asynchronously to the coupling facility (CF).
Field Name: QTGAFLSE

FALSE CONTENTIONS
The total number of false contentions for LOCK and UNLOCK requests. A false contention occurs when different resource names hash to the same entry in the coupling facility (CF) lock table. The CF detects contention within the hash entry, and XES uses intersystem messaging to determine that no actual resource contention exists.
Field Name: QTGAFCNT
Record trace - IFCID 003 - DDF Data by Location

The field labels shown in the following sample layout of “Record Trace - IFCID 003 - DDF Data by Location” are described in the following section.

**DDF DATA BY LOCATION**

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>TYPE</td>
<td>PACKAGE</td>
</tr>
<tr>
<td>ROLLED NBR THREADS</td>
<td>1</td>
</tr>
<tr>
<td>LOCATION NAME (LONG)</td>
<td>::FFFF:9.65.12.94</td>
</tr>
<tr>
<td>LOCATION NAME (SHORT)</td>
<td>::FFFF:9.65.12.9</td>
</tr>
<tr>
<td>FLAGS</td>
<td>X'00'</td>
</tr>
<tr>
<td>ABDRT REQUESTS RECEIVED</td>
<td>0</td>
</tr>
<tr>
<td>BLKS RCVD USING BLK FETCH.</td>
<td>0</td>
</tr>
<tr>
<td>BYTES RCVD FROM REMOTE</td>
<td>0</td>
</tr>
<tr>
<td>CONV REQs QUEUED BY DDF</td>
<td>0</td>
</tr>
<tr>
<td>CONV INITIATED FR LOCAL</td>
<td>0</td>
</tr>
<tr>
<td>COMMIT REQs RECVD FR COORD.</td>
<td>1</td>
</tr>
<tr>
<td>MSGs RCVD FR REMOTE</td>
<td>8</td>
</tr>
<tr>
<td>ROWs OF DATA RETR FR REMOTE</td>
<td>0</td>
</tr>
<tr>
<td>SQL STMT RECVD FR REMOTE</td>
<td>3</td>
</tr>
<tr>
<td>LOCAL ELAPSED TIME</td>
<td>0.000000</td>
</tr>
<tr>
<td>BACKOUT REQs RECVD FR COORD.</td>
<td>N/A</td>
</tr>
<tr>
<td>ROWs MSG BUFFER BLK FETCH.</td>
<td>N/A</td>
</tr>
<tr>
<td>LARGEST CV(ALLOC-DEALLOC)</td>
<td>N/A</td>
</tr>
<tr>
<td>COMMITS USING REM LOCATION.</td>
<td>N/A</td>
</tr>
<tr>
<td>COMMIT REQs RECVD FR COORD.</td>
<td>N/A</td>
</tr>
<tr>
<td>AGNT CPU TIME REMOTE SITE.</td>
<td>N/A</td>
</tr>
<tr>
<td>LAST AGNT REQs RECVD FR INIT.</td>
<td>N/A</td>
</tr>
<tr>
<td>PREPARE REQs RECVD FR COORD.</td>
<td>N/A</td>
</tr>
<tr>
<td>SQL STMTS BOUND FOR REM ACC.</td>
<td>N/A</td>
</tr>
<tr>
<td>THREAD ALLOC REQs RECEIVED.</td>
<td>N/A</td>
</tr>
<tr>
<td>BACKOUT RESP RECVD FR PART.</td>
<td>N/A</td>
</tr>
<tr>
<td>COMMIT RESP RECVD FR PART.</td>
<td>N/A</td>
</tr>
</tbody>
</table>

**TYPE (Either ROLLUP, ROLSUM OR PACKAGE)**

The flag byte:

- **X'20'**  This value is shown if DRDA is used to communicate with the server.
- **X'40'**  This value is shown if DB2 private protocol is used to communicate with the server.

All other values shown in this field are serviceability.

**Field Name: QLACFLGS**

**LOCATION NAME (LONG)**

The name of the remote location with which this information is associated.

If the local location is the requester, this field is a server location. If the local location is a server location, this field is the requester location. An allied thread is created at a DB2 requester, and a database access thread is created at a DB2 server. An accounting record is for either a requester or a server, but not for both.

This field is invalid in case of DB2 10 or later if summary rollup data is present. In Accounting this field is set to "ROLSUM".

**Field Name: QLACLOCN**

This is an exception field.

**LOCATION NAME (SHORT)**

The name of the remote location with which this information is associated.

If the local location is the requester, this field is a server location. If the
local location is a server location, this field is the requester location. An allied thread is created at a DB2 requester, and a database access thread is created at a DB2 server. An accounting record is for either a requester or a server, but not for both.

This field is invalid in case of DB2 10 or later if summary rollup data is present. In Accounting this field is set to "ROLSUM".

Field Name: QLACLOCN
This is an exception field.

ABORT REQUESTS RECEIVED
This field depends on the DB2 version that is installed:
- **DB2 10 or later**: The number of abort requests received from the requester (single-phase commit protocol) and backout requests received from the coordinator (two-phase commit protocol).
- **Prior to DB2 10**: The number of rollback requests received from the requester location (single-phase commit operations only). This value is maintained at the server location.

Field Name: QLACABRR
This is an exception field.

BLKS RECV USING BLK FETCH
The number of blocks received using block fetch. This value is maintained at the requester location.

Field Name: QLACBRBF
This is an exception field.

BYTES RECV FROM REMOTE
The number of bytes the server location received from the requester location.

More bytes of data might be sent from the server location than are received by the requester, because of the way in which distributed SQL statements are processed internally.

Field Name: QLACBYTR
This is an exception field.

CONV REQS QUEUED BY DDF
A number of conversation requests queued by DDF that are waiting for allocation. This value is maintained at the requester location.

If the value is a large number, you might want to increase the limit for the number of conversations.

Field Name: QLACCNVQ
This is an exception field.

CONV INITIATED FR LOCAL
The number of conversations (both successful and unsuccessful) initiated by the requester location to be executed at the server location. This number is maintained at the requester.

Field Name: QLACCNVS
This is an exception field.

COMMIT REQS RECV FR REQ/COO
This field depends on the DB2 version that is installed:
• DB2 10 or later: The number of commit requests received from the requester (single-phase commit protocol) and committed requests received from the coordinator (two-phase commit protocol).
• Prior to DB2 10: The number of single-phase commit requests received from the requester location. This value is maintained at the server location.

Field Name: QLACCOMR
This is an exception field.

MSGS RECV FR REMOTE
The number of messages received from the location. This value is maintained at the location where the messages were received.

More messages might be sent from the server location than are received by the requester because of the way in which distributed SQL statements are processed internally.

Field Name: QLACMSGR
This is an exception field.

ROWS OF DATA RETR FR REMOTE
The number of rows of data retrieved from the server location. This value is maintained at the requester location.

Special Considerations:
1. The number of rows received from the server location does not include either the SQLDA or SQLCA.
2. Block fetch can significantly affect the number of rows sent across the network. When used with non-UPDATE cursors, block fetch puts as many rows as possible into the message buffer, and transmits the buffer across the network without requiring a VTAM message. Consequently, more rows of data might be sent from the server location than are received by the reporting (requester) location. This is especially true when DB2 private protocol is used because multiple blocks can be transmitted from the server with no intervening messages sent by the requester.

Field Name: QLACROWR
This is an exception field.

SQL STMT RECEIVED FR REMOTE
The number of SQL statements received from the requester location.

Field Name: QLACSQLR
This is an exception field.

LOCAL ELAPSED TIME
The elapsed time at the requester location until the database access agent completed its work, including DB2 processing time and network time. This
value is maintained at the requester location and is calculated by
accumulating the difference between the store clock values obtained before
and after each network request.

Field Name: QLACCPUL

BACKOUT REQs RECv FR CoORD

The number of backout requests received from the coordinator (two-phase
commit operations only). This value is maintained at the participant.

Field Name: QLACBKRC

ROWS MSG BUFFER BLK FET

The number of rows transmitted or received in DB2 message buffers using
block fetch. This includes both requester and server activity.

Field Name: QLACBROW

This is an exception field.

LARGEST CV(ALLOC-DEALLOC)

The maximum number of conversations open at any time (QLACCNVAT -
QLACCNVIT). QLACIE is updated only when (QLACCNVAT -
QLACCNVIT) is greater than the current value of QLACIE. QLACF1
and QLACF2 indicate whether the conversations use DB2 private
protocol, DRDA protocol, or both. This value is maintained at the requester
location.

Field Name: QLACIE

This is an exception field.

COMMITS USING REM LOCATION

The number of commit operations performed with the remote location as
coordinator (two-phase commit operations only). It is maintained at the
participant.

Field Name: QLACCPTR

This is an exception field.

COMMIT REQs RECv FR CoORD

The number of commit requests received from the coordinator (two-phase
commit operations only). This value is maintained at the participant,
indicating that the participant was read only.

Field Name: QLACCRRC

This is an exception field.

AGNT CPU TIME REMOTE SITE

The database access agent CPU time at the server location. This time does
not include most of the time spent in VTAM and is only reported for DB2
private protocol requests. If the agent uses both methods to communicate
with the remote location, only the CPU time associated with the DB2
private protocol is reported, and this time can be misleading. If only
DRDA is used, this value is 0.

Field Name: QLACDBAT

LAST AGNT REQs RECv FR INIT
The number of last agent requests received from the initiator (two-phase commit operations only). It is maintained at the participant.

Field Name: QLaCLARC

PREPARE REQS RECEIVED FROM COORD

The number of PREPARE requests received from the coordinator (two-phase commit operations only). It is maintained at the participant.

Field Name: QLaCPRRRC

FORGET RESP RECEIVED FROM PART

The number of forget responses received from the participant (two-phase commit operations only). It is maintained at the coordinator.

Field Name: QLaCRCRRRC

SQL STMS BOUND FOR REM ACC

The number of static SQL statements that were bound for remote access (DB2 private protocol only). This value is maintained at the requester location.

Field Name: QLaCRBND

THREAD ALLOC REQS RECEIVED

The number of CREATE DATABASE ACCESS THREAD (DBAT) requests received by the server DBAT from the requester allied agent. This number is maintained by the server DBAT and is always 1.

Field Name: QLaCTRNR

BACKOUT RESP RECEIVED FROM PART

The number of backout responses received from the participant (two-phase commit operations only). It is maintained at the coordinator and indicates that the participant rejected the PREPARE request.

Field Name: QLaCVNRRC

COMMIT RESP RECEIVED FROM PART

The number of request commit responses received from the participant (two-phase commit operations only). It is maintained at the coordinator.

Field Name: QLaCVYRC

ROLLED NBR THREADS

The number of threads to roll data into this QLAC data section. Non-rollup QLACs have a value of 1 and rollup QLACs have a value of 1 or more.

Field Name: QLaCRLNU

FLAGS

The flag byte:

X'20' This value is shown if DRDA is used to communicate with the server.

X'40' This value is shown if DB2 private protocol is used to communicate with the server.

All other values shown in this field are serviceability.

Field Name: QLaCFLGS
ABORT REQUESTS SENT

This field depends on the DB2 version that is installed:

- **DB2 10 or later:** The number of abort requests sent to the server (single-phase commit protocol) and backout requests sent to the participant (two-phase commit protocol).
- **Prior to DB2 10:** The number of rollback requests sent to the server location (single-phase commit operations only). This value is maintained at the requester location.

**Field Name:** QLACABRS

This is an *exception* field.

BLKS TRANS USING BLK FETCH

The number of blocks transmitted using block fetch. This value is maintained at the server location.

**Field Name:** QLACBTBF

This is an *exception* field.

BYTES SENT TO REMOTE

The number of bytes the server location sent to the requester location. This value is maintained at the server location.

More bytes of data might be sent from the server location than are received by the requester due to the way in which distributed SQL statements are processed internally.

**Field Name:** QLACBYTS

This is an *exception* field.

CONV INITIATED FR REMOTE

A count of conversations initiated by the requester.

This number is updated at the server location.

**Field Name:** QLACCNVR

This is an *exception* field.

CONV TERMINATED FR LOCAL

The number of terminated conversations in the server block (DB2 private protocol only). It is maintained at the requester location.

This number can be different from the number of successful conversation allocations, because some conversations might not have been terminated when the accounting record was written.

**Field Name:** QLACCNVT

This is an *exception* field.

COMMIT REQS SENT TO SRV/PAR

This field depends on the DB2 version that is installed:

- **DB2 10 or later:** The number of commit requests sent to the server (single-phase commit protocol) and committed requests sent to the participant (two-phase commit protocol).
- **Prior to DB2 10:** The number of single-phase commit requests sent to the server location. This value is maintained at the requester location.
Field Name: QLACCOMS
This is an exception field.

**MSGS SENT TO REMOTE**
The number of messages sent to the location. It is maintained at the location where the messages originated.

Field Name: QLACMSGS
This is an exception field.

**ROWS OF DATA SENT TO REMOTE**
The number of rows sent from the server location to the requester location. The value includes SQLDA and is maintained at the server location.

Field Name: QLACROWS
This is an exception field.

**SQL STMT SENT TO REMOTE**
The number of SQL statements sent to the server location. This value is maintained at the requesting location.

Field Name: QLACSQLS
This is an exception field.

**INDOUBT THREADS**
The number of threads that went indoubt with the remote location as coordinator (two-phase commit operations only). It is maintained at the participant and indicates that the communication with the coordinator was lost.

Field Name: QLACINDT
This is an exception field.

**BACKOUT REQS SENT TO PART**
The number of backout requests sent to the participant (two-phase commit operations only).

Field Name: QLACBKSE
This is an exception field.

**SWITCH CONT BLK TO LIM BLK**
The number of times continuous block mode switched to limited block mode (DB2 private protocol only).

Field Name: QLACCBLB
This is an exception field.

**SUCC CONV ALLOCATED**
The number of successful conversation allocations made to the server (DB2 private protocol only). This value is maintained at the requester location.

All allocation attempts, whether successful or not, are counted in QLACCNVS. The difference between QLACCNVS and this field helps to identify session resource constraint problems. Counting the number of unsuccessful conversations is useful for session tuning.
Field Name: QLACCNVA

ELAPSED DB AGENT TIME

The elapsed database access agent time at the server location. This time is updated at the requester location and is reported only for DB2 private protocol. If the agent uses both methods to communicate with the server location, then only the elapsed time associated with the DB2 private protocol is reported, and this time can be misleading. If only DRDA is used, this value is 0.

Field Name: QLACCPUR

COMMIT REQS SENT TO PART

The number of commit requests sent to the participant (two-phase commit operations only). This value is maintained at the participant, indicating that the participant was read only.

Field Name: QLACCRSE

PRID REMOTE SITE

The product ID and version of the remote location.

This field is invalid:
- In Accounting trace, it shows N/P.
- In Accounting report, it shows the last product ID being reduced, or hexadecimal 0 in case of DB2 10 or later and rollup summary data.
- In Accounting FILE and SAVE PROGRAM table, it shows blank.
- In case of DB2 10 or later if summary rollup data is present.

Field Name: QLACPRID

LAST AGNT REQS SENT TO COOR

The number of last agent requests sent to the coordinator (two-phase commit operations only).

A last agent request reduces the number of messages that must be sent for the commit. If DB2 is the requester, this number is incremented when a conversation is deallocated and this conversation was not used since the last commit. If this number is large, and your application design permits it, you can save another message by issuing a release before the commit (only for a DB2 requester).

Field Name: QLACLASE

PREPARE REQS SENT TO PART

The number of PREPARE requests sent to the participant (two-phase commit operations only). It is maintained at the coordinator.

Field Name: QLACPRSE

FORGET RESP SENT TO COORD

The number of forget responses sent to the coordinator (two-phase commit operations only). It is maintained at the participant.

Field Name: QLACRRSE

ROLLBACKS REM SITE AS COORD
ICCID 003 - DDF Data by Location

The number of rollback operations performed with the remote location as coordinator (two-phase commit operations only). It is maintained at the participant.

**Field Name:** QLACRBTR

**THREAD ALLOC REQS SENT**

The number of CREATE DATABASE ACCESS THREAD (DBAT) requests the requester allied agent sent to the server location. This number is maintained by the requester allied agent.

In some cases, for example when a new user signs on or a resignon occurs, the value of this field can be zero. This indicates that the existing DBAT at the server was reused by this user.

**Field Name:** QLACTRNS

**BACKOUT RESP SENT TO COORD**

The number of backout responses sent to the coordinator (two-phase commit operations only). It is maintained at the participant and indicates that the participant rejected the PREPARE request.

**Field Name:** QLACVNSE

**COMMIT RESP SENT TO COORD**

The number of request commit responses sent to the coordinator (two-phase commit operations only). It is maintained at the participant.

**Field Name:** QLACVYSE
IFCID 003 - Dynamic SQL Statement

This topic shows detailed information about “Record Trace - IFCID 003 - Dynamic SQL Statement”.

Record trace - IFCID 003 - Dynamic SQL Statement

The field labels shown in the following sample layout of “Record Trace - IFCID 003 - Dynamic SQL Statement” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>DYNAMIC SQL STMT</td>
<td></td>
</tr>
<tr>
<td>REOPTIMIZATION</td>
<td>0</td>
</tr>
<tr>
<td>STMT INVALID (MAX)</td>
<td>0</td>
</tr>
<tr>
<td>STMT INVALID (DML)</td>
<td>0</td>
</tr>
<tr>
<td>CSWL STMTS PARSED</td>
<td>0</td>
</tr>
<tr>
<td>CSWL DUPLS CREATED</td>
<td>0</td>
</tr>
<tr>
<td>FOUND IN CACHE</td>
<td>0</td>
</tr>
<tr>
<td>IMPLICIT PREPARES</td>
<td>10</td>
</tr>
<tr>
<td>NOT FOUND IN CACHE</td>
<td>0</td>
</tr>
<tr>
<td>PREPARES AVOIDED</td>
<td>0</td>
</tr>
<tr>
<td>STMT INVALID (DDL)</td>
<td>0</td>
</tr>
<tr>
<td>CSWL LITS REPLACED</td>
<td>0</td>
</tr>
<tr>
<td>CSWL MATCHES FOUND</td>
<td>0</td>
</tr>
</tbody>
</table>

REOPTIMIZATION

The total number of times reoptimization occurs because the value of the host variable or parameter marker changes.

Field Name: QXSTREOP

FOUND IN CACHE

The number of times a PREPARE command was satisfied by copying a statement from the prepared statement cache.

Field Name: QXSTFND

IMPLICIT PREPARES

An implicit prepare occurs when the user copy of the prepared SQL statement no longer exists in the local dynamic SQL cache and the application plan or package is bound with KEEPDYNAMIC YES.

If the skeleton copy of the prepared SQL statement exists in the global dynamic SQL cache in the EDM pool, a short prepare is executed, otherwise a full prepare is executed.

Field Name: QXSTIPRP

STMT INVALID (MAX)

The number of times statements are invalidated in the local dynamic SQL cache because the MAXKEEPD limit has been reached and prepared SQL statements in the local dynamic SQL cache have to be reclaimed.

Field Name: QXSTDEXP

NOT FOUND IN CACHE

The number of times that DB2 searched the prepared statement cache but could not find a suitable prepared statement.

Field Name: QXSTNFND

PREPARES AVOIDED

This field indicates the number of times where no SQL PREPARE or EXECUTE IMMEDIATE was issued by the application and a copy of a prepared SQL statement was found in local dynamic SQL cache.

When an application plan or package is bound with KEEPDYNAMIC YES, a copy of each prepared SQL statement for the application thread is held in the local dynamic SQL cache and kept across a commit boundary.
An application thread can save the total cost of a prepare by using a copy of the prepared statement in the local dynamic SQL cache from an earlier prepare by the same thread. To do this, the application must be modified to avoid issuing repetitive SQL PREPAREs for the same SQL statement.

Field Name: QXSTNPRP

STMT INVALID (DDL)
The number of times statements are invalidated in the local dynamic SQL cache because of SQL DDL or updated RUNSTATS information and prepared SQL statements in the local dynamic SQL cache have to be reclaimed.

Field Name: QXSTDINV

CSWL STMTS PARSED
The number of times DB2 parsed dynamic statements because CONCENTRATE STATEMENTS WITH LITERALS behavior was used for the prepare of the statement for the dynamic statement cache.

Field Name: QXSTCWLP

CSWL LITS REPLACED
The number of times DB2 replaced at least one literal in a dynamic statement because CONCENTRATE STATEMENTS WITH LITERALS was used for the prepare of the statement for dynamic statement cache.

Field Name: QXSTCWLR

CSWL MATCHES FOUND
The number of times DB2 found a matching reusable copy of a dynamic statement in cache during prepare of a statement that had literals replaced because of CONCENTRATE STATEMENTS WITH LITERALS.

Field Name: QXSTCWLM

CSWL DUPLS CREATED
The number of times DB2 created a duplicate STMT instance in the statement cache for a dynamic statement that had literals replaced by CONCENTRATE STATEMENTS WITH LITERALS behavior. The duplicate STMT instance was needed because a cache match failed because the literal reusability criteria was not met.

Field Name: QXSTCWLD
IFCID 003 - Group Buffer Pools Activity Data

This topic shows detailed information about “Record Trace - IFCID 003 - Group Buffer Pools Activity Data”.

Record trace - IFCID 003 - Group Buffer Pools Activity Data

The field labels shown in the following sample layout of “Record Trace - IFCID 003 - Group Buffer Pools Activity Data” are described in the following section.

GROUP BUFFER POOLS ACTIVITY DATA

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>GROUP BUFFER POOL ID</td>
<td>0</td>
</tr>
<tr>
<td>READ(NF)-DATA RETURNED</td>
<td>0</td>
</tr>
<tr>
<td>READ(XI)-DATA RETURNED</td>
<td>0</td>
</tr>
<tr>
<td>READ(XI)-NO DATA RET.</td>
<td>0</td>
</tr>
<tr>
<td>EXPLICIT X-INVALID</td>
<td>0</td>
</tr>
<tr>
<td>WRITE TO SEC GBP</td>
<td>0</td>
</tr>
<tr>
<td>ASYNCH. GBP REQUESTS</td>
<td>23</td>
</tr>
<tr>
<td>ASYNCH. SEC-GBP REQUESTS</td>
<td>24</td>
</tr>
</tbody>
</table>

GROUP BUFFER POOL ID

The group buffer pool identifier.

Field Name: QBGAGN

READ(NF)-DATA RETURNED

The number of coupling facility read requests necessary because the requested page was not found in the buffer pool. Data is returned from the coupling facility.

Field Name: QBGAMD

CLEAN PAGES WRITTEN

The number of clean pages written to the group buffer pool.

Field Name: QBGAWC

READ(XI)-DATA RETURNED

The number of coupling facility read requests required because the buffer was marked invalid. Data is returned from the group buffer pool.

Field Name: QBGAXD

READ(NF)-NO DATA RET

The number of group-buffer-pool reads due to local buffer-pool miss where no data was returned.

Field Name: QBGAMR

CHANGED PAGES WRITTEN

The number of changed pages written to the group buffer pool as a result of write and register (WAR), or write and register multiple (WARM) requests.

Field Name: QBGASW

This is an exception field.

READ(XI)-NO DATA RET

The number of group buffer pool read requests due to buffer XI where no data was returned.

Field Name: QBGAXR

READ PREFETCH
### IFCID 003 - Group Buffer Pools Activity Data

The number of pages read from the group buffer pool due to prefetch under the control of the agent.

**Field Name:** QBGAMN

**UNREGISTER PAGE**

The number of coupling facility requests to unregister a page.

**Field Name:** QBGADG

**EXPLICIT CROSS-INVALIDATIONS**

The number of times an explicit coupling facility cross-invalidation request was issued.

**Field Name:** QBGAEX

**WRITE TO SEC-GBP**

The number of requests to write changed pages to the secondary GBP for duplexing.

**Field Name:** QBGA2W

**ASYNCH. GBP REQUESTS**

The number of asynchronous IXLCACHE invocations for the primary group buffer pool.

**Field Name:** QBGAHS

**ASYNCH. SEC-GBP REQUESTS**

The number of IXLCACHE invocations for the secondary group buffer pool.

**Field Name:** QBGA2H
IFCID 003 - IFI Class 5 Times and Data Capture

This topic shows detailed information about “Record Trace - IFCID 003 - IFI Class 5 Times and Data Capture”.

Record trace - IFCID 003 - IFI Class 5 Times and Data Capture

The field labels shown in the following sample layout of “Record Trace - IFCID 003 - IFI Class 5 Times and Data Capture” are described in the following section.

### IFCID 003 - IFI Class 5 Times and Data Capture

#### IFI CALL ELAPSED TIME

The accumulated elapsed time for processing IFI calls. This field is only calculated if accounting class 5 is active.

**Field Name:** QIFAAIET

#### IFI CALL TCB CPU TIME

The accumulated CPU time spent processing IFI calls. This is the same as the TCB time (class 5).

This field is only calculated if accounting class 5 is active.

**Field Name:** QIFAAITT

This is an exception field.

#### DESCRIBES ELAPSED

The accumulated elapsed time for processing data capture describes. Data capture describes occur only during IFI read requests for IFCID 185. This time is a subset of the log extraction time.

**Field Name:** QIFAAMBT

This is an exception field.

#### LOG EXTRACT ELAPSED

The accumulated elapsed time for extracting log records for tables defined with DATA CAPTURE CHANGES. This time is a subset of the class 5 elapsed time.

**Field Name:** QIFAAMLT

This is an exception field.

#### IFI CALLS

The number of IFI calls.

**Field Name:** QIFAANIF

#### LOG READS PERFORMED

The number of log reads performed for processing IFI READS requests for IFCID 185.

**Field Name:** QIFAANLR

#### LOG RECS CAPTURED

The number of retrievable log records that were written for tables defined with DATA CAPTURE CHANGES. This number includes only those log
IFCID 003 - IFI Class 5 Times and Data Capture

records that can be retrieved by an IFI READS call for IFCID 185. Some records can be written but not retrieved, for example if monitor trace class 6 is not active.

Field Name: QIFAANRC

DATA DESCR. RETURNED

The number of data descriptions returned in IFCID 185. The data descriptions are mapped in IFCID 185.

Field Name: QIFAANDD

DESCRIPTES

The number of data capture describes for processing READS requests for IFCID 185 data.

Field Name: QIFAANMB

DATA ROWS RETURNED

The number of data rows returned in IFCID 185. Two rows are returned for each row altered by an SQL UPDATE statement.

Field Name: QIFAANDR

LOG RECS RETURNED

The number of log records returned to the caller of the IFI READS call for IFCID 185.

Field Name: QIFAANRR

TABLES RETURNED

The total number of tables returned to the caller of IFI READS call for IFCID 185.

Field Name: QIFAANTB
Record trace - IFCID 003 - Initial DB2 Requester and MVS Correlation Data

The field labels shown in the following sample layout of “Record Trace - IFCID 003 - Initial DB2 Requester and MVS Correlation Data” are described in the following section.

**BYTES**

The length of the product ID and accounting string.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

Field Name: QMDAASLN

**PRODUCT ID**

Shows the product identifier (ID) of the requester. It can have the following values:

- **DB2** For DB2 UDB for z/OS
- **SQL/DS** For DB2 UDB for VSE and VM
- **JDBC DRIVER** For Universal JDBC driver
- **COMMON SERV** For DB2 UDB for Linux, UNIX, Windows
- **DB2/400** For DB2 UDB for iSeries

Otherwise, it shows the first 3 characters of the product ID, or N/P if the record was written at the application requester location.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

Field Name: QMDAPRID

**PRODUCT VERSION**

The version, release, and modification level of the product, which generated the accounting information. It has the following format: \( vv \ rr \ m \), where:

- \( vv \) Version level
- \( rr \) Release level
- \( m \) Modification level

N/P is shown if the record was written at the application requester location.
**IFCID 003 - Initial DB2 Requester and MVS Correlation Data**

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

**Field Name:** QMDAPMOD

**LOCATION**

The location name for the DB2 subsystem that created the QMDAINFO values.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

**Field Name:** QMDALOCN

**NET ID**

The NETID of the DB2 subsystem that created the QMDAINFO values.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

**Field Name:** QMDANETN

**LU NAME**

The SNA LU name of the DB2 subsystem that created the QMDAINFO values.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

**Field Name:** QMDALUNM

**CONNECT**

The connection name of the DB2 system that created the MVS and DDF accounting values.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

**Field Name:** QMDACNAM

**CONNTYPE**

The type of subsystem connection at the DB2 system where the SQL application is running. Possible values and their descriptions are:

<table>
<thead>
<tr>
<th>BATCH</th>
<th>TSO or call attach</th>
</tr>
</thead>
<tbody>
<tr>
<td>SASS</td>
<td>CICS</td>
</tr>
<tr>
<td>MASS</td>
<td>IMS</td>
</tr>
<tr>
<td>DIST</td>
<td>Distributed</td>
</tr>
</tbody>
</table>

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

**Field Name:** QMDACTYP

**CORRNAME**

The first 8 bytes of the correlation ID at the DB2 system running the SQL.

The last 4 bytes of the correlation ID at the DB2 system running the SQL.
For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

Field Name: QMDACORR

AUTHID

The DB2 authorization ID that the SQL application used before name translation and before driving the connection exit at the DB2 site where the SQL application is running.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

Field Name: QMDAAUTH

PLANNNAME

The DB2 plan used at the DB2 system running the SQL.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

Field Name: QMDAPLAN

MVS ACCOUNTING DATA

The MVS accounting string associated with the MVS address space of the SQL application. It is filled if PRD_TYP=D; otherwise X'00' is used.

This information comes from the ACCT= parameter on the job statement. If the ACCT= parameter is blank, the information on the EXEC statement is used. TSO logon Accounting information is used only if there is a value in the account field on the TSO Logon panel. Do not confuse this field with the Accounting correlation token.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

Field Name: QMDAACCT

CLIENT PLATFORM

The client platform, such as AIX. This is a 1 to 18 character field padded with blanks.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

Field Name: QMDAPLAT

CLIENT APPLICATION NAME

The name of the client application. This is a 1 to 20 character field padded with blanks. An example is "PAYROLL".

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

Field Name: QMDAAPPL

CLIENT AUTHID

The client authorization ID of an application process. This is a 1 to 8 character field padded with blanks. An example is "SMITH".

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.
IFCID 003 - Initial DB2 Requester and MVS Correlation Data

Field Name: QMDAATID

DDCS ACCOUNT SUFFIX

The account suffix. The maximum length of this field is 200 bytes. This field is the user-supplied portion (suffix) of the accounting string. An example is "DEFAULT_DRDA". A value of zero in QMDASFLN Indicates there is no account suffix.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

Field Name: QMDASUFX

ACCOUNTING STRING

The accounting string:

- For local DB2 threads, the format of the accounting string is shown in QMDAINFO.
- For database access threads, the accounting string contains the accounting string sent by the requester.
- The QMDAPRID value identifies which product generated the accounting string.
  - If the requester is DB2, the accounting string is defined in QMDAINFO.
  - If QMDAPTY is DSN, QMDAINFO defines the format.
  - If QMDAPTY is SQL or JCC, QMDASQLI defines the format.
  - Otherwise, the format is undefined.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup.

Field Name: QMDAASTR
## Record Trace - IFCID 003 - Instrumentation Accounting Data

The field labels shown in the following sample layout of “Record Trace - IFCID 003 - Instrumentation Accounting Data” are described in the following section.

### Class 1: Beginning Store Clock Time

The beginning store clock value for the period covered by the accounting record. You can determine the elapsed time of the application by subtracting this field from the ending store clock value (QWACESC). Threads that do not terminate (such as CICS primed threads and IMS wait-for-input message regions) can have an ending clock value that includes the time during which the thread was inactive and waiting for work.

If a roll-up trace record is written with accumulated counter data, QWACBSC represents the earliest begin store clock value for a thread that has rolled data into the record. In this case, QWACESC shows the accumulated elapsed time.
**Field Name:** QWACBSC

**CLASS 1: ENDING STORE CLOCK TIME**

The ending store clock value. You can use this field with the beginning store clock value (QWACBSC) to determine the elapsed time of an application.

If a roll-up record is written with accumulated accounting data, QWACESC contains the accumulated elapsed time. In Accounting Trace reports, the elapsed time is shown under CLASS 1: NONNESTED ELAPSED TIME and the END TIME is reported as N/P, because QWACESC does not contain a timestamp. In the Accounting FILE GENERAL table, the accumulated elapsed time QWACESC is stored in column CLASS1_ELAPSED and column CLASS1_TIME_END contains a timestamp 1900-01-01-00.00.00.000000.

**Field Name:** QWACESC

**CLASS 1: ELAPSED TIME**

The time covered by this accounting record. If the time cannot be calculated or the value is negative, N/C is printed in this field. Calculated from the DB2 field QWACESC - QWACBSC.

**Field Name:** RT0003ET

**CLASS 1: MVS TCB TIME**

The amount of MVS CPU time used. If the time cannot be calculated or the value is negative, N/C is printed in this field. Calculated from the DB2 field QWACEJST - QWACBJST.

**Field Name:** RT0003TT

**CLASS 1: BEGINNING MVS TCB TIME**

The beginning MVS CPU time for all environments (such as: CICS, IMS, RRSAF, or TSO). This CPU time is not affected by an IBM specialty engine. Binary zero means that no time value is available.

**Field Name:** QWACBJST

**CLASS 1: ENDING MVS TCB TIME**

The ending MVS CPU time. This CPU time is not affected by an IBM specialty engine. Binary zero means that no time value is available.

**Field Name:** QWACEJST

**CLASS 1: STORED PROC ELAPSED TIME**

The total elapsed time spent by the allied agent in stored procedures. A stored procedure may initiate a trigger or invoke a user-defined function. The time spent there is not included in this counter.

**Note:** This field is not normally shown in the short layouts but can be included with UTR.

**Field Name:** QWACSPEA

**CLASS 1: CONVERSION FACTOR**

The CPU service unit conversion factor allows for converting CPU time to a common unit, which is called service unit (SU). The conversion factor depends on the machine being used. With the SU, you can add up CPU
IFCID 003 - Instrumentation Accounting Data

Execution times across multiple DB2 systems running on different machines. It is a raw value for RECORDTRACE and Accounting FILE data. For Accounting SAVE data it cannot be determined.

Field Name: QWACSUCV

CLASS 1: STORED PROCEDURE TCB TIME

The TCB time accumulated in DB2 for processing SQL CALL statements in the stored procedures or WLM address space. This time is only calculated if accounting class 1 is active.

SQLP times are included in this time if the SQLP was called on a nested task and was not invoked by the main application execution unit. This time does not include CPU time consumed on an IBM specialty engine.

Field Name: QWACSPCP

CLASS 1: PAR. TASKS

The number of parallel child agents, or Accounting intervals rolled up, or autonomous procedures rolled up. The value depends on the record type:

- For a non-rollup parent record, this value is the number of parallel child agents that were created.
- For a non-rollup child agent record, this value is 0.
- For a parallel query rollup record, this value is the number of parallel child agents rolled into the record.
- For a DDF/RRSAF rollup record, this value is the number of Accounting intervals that were rolled into the record for the corresponding end user.
- For an autonomous procedure rollup record, this value is the number of autonomous procedures rolled into the record.

Field Name: QWACPCNT

CLASS 1: PAR. TOKEN

Token used to correlate parallel task, utility task records, or autonomous transaction rollup records with the records of the originating task or main utility task.

Field Name: QWACPACE

CLASS 1: UDF ELAPSED TIME

The total elapsed time spent by the allied agent in UDF functions processed in a DB2 stored procedure or WLM address space. A user-defined function may initiate a trigger or invoke a stored procedure. Non-inline UDF times are included in this time if the native UDF was called on a nested task and was not invoked by the main application execution unit.

This time includes time executing SQL.

Note: With user-tailored reporting (UTR) you can include this field in the short layouts of Accounting.

Field Name: QWACUDEA

CLASS 1: COMMITS

The number of successful two-phase (units of recovery) or single-phase (syncs) commit requests. It indicates the number of units of recovery that are completed successfully, and for which the associated commit duration
locks were released. It represents the total number of commit requests processed by the DB2 subsystem, whether the request was an explicit or implicit external request from an IMS or a CICS connection, or an implicit internal request within DB2 when DB2 was the commit coordinator or conducted read-only commit processing as a commit participant on phase-1 calls from an IMS or CICS connection.

For parallel queries, only the commits from the initiating (parent) thread are recorded by this counter.

Field Name: QWACCOMM

This is an exception field.

CLASS 1: SVPT REQ.

The number of named SAVEPOINTs set within a transaction.

Field Name: QWACSVPT

CLASS 1: CP CPU TIME UDF

The accumulated CPU time used to satisfy UDF requests processed in a DB2 stored procedure or WLM address space. Non-inline UDF times are included in this time if the native UDF was called on a nested task and was not invoked by the main application execution unit.

This time is only calculated if accounting class 1 is active.

This time does not include the CPU time consumed on an IBM specialty engine.

Field Name: QWACUDCP

CLASS 1: ROLLBACKS

The number of rollback requests. This is the number of units that were backed out, including rollbacks from attaches.

Special Considerations: This field contains the number of:
- Application program abends
- Application rollback requests
- Application deadlocks on database records
- Applications canceled by operator
- Thread abends due to resource shortage

Field Name: QWACABRT

This is an exception field.

CLASS 1: SVPT RLB.

The number of ROLLBACK TO SAVEPOINT statements executed.

Field Name: QWACRBSV

CLASS 1: NETWORK ID VALUE

The network ID. It is used with IMS and CICS.

Field Name: QWACNID

CLASS 1: PROGRAMS

The number of packages or DBRMs for which accounting data was collected.
IFCID 003 - Instrumentation Accounting Data

Field Name: QWACPKGN

CLASS 1: SVPT REL.
The number of RELEASE SAVEPOINT statements executed.

Background and Tuning Information
Release savepoints as soon as possible. Outstanding savepoints block SQL operations that resolve remote locations. DB2 always releases outstanding savepoints when a transaction ends.

Field Name: QWACRLSV
This is an exception field.

CLASS 1: REASON ACCT INVOKED
The reason for termination, that is, for producing a DB2 accounting record.

Field Name: QWACRINV
This is an exception field.

CLASS 1: SE CPU TIME
The accumulated CPU time that is consumed while running on an IBM specialty engine in all environments. This value may be 0 when QWACRINV is greater than or equal to 20.

Field Name: QWACCLS1_ZIIP

CLASS 1/2 STORED PROC SE TCB TIME
The accumulated CPU time that is consumed while running stored procedure requests on the main application execution unit on an IBM specialty engine. As these SPs run entirely within DB2, this time represents class 1 and class 2 time.

Field Name: QWACSPNF_ZIIP

CLASS 1/2 STORED PROC ELAPSED TIME
The accumulated elapsed time that is consumed on an IBM specialty engine for executing stored procedure requests on the main application execution unit. As these stored procedures run entirely in DB2, this time represents class 1 and class 2 time.

Field Name: QWACSPNF_ELAP

CLASS 1/2 STORED PROC CP ELAPSED TIME
The accumulated CPU time that is used for executing stored procedure requests on the main application execution unit. This time does not include the time that is consumed on an IBM specialty engine. As these stored procedures run entirely in DB2, this time represents class 1 and class 2 time.

Field Name: QWACSPNF_CP

CLASS 1/2 UDF NF SE CPU TIME
Accumulated CPU time consumed executing user-defined functions on the main application execution unit on an IBM specialty engine. Since these UDFs run entirely within DB2, this time represents class 1 and class 2 time.

Field Name: QWACUDFNF_ZIIP

CLASS 1/2 UDF NF ELAPSED TIME

Chapter 40. IFCID Record Blocks 40-179
IFCID 003 - Instrumentation Accounting Data

Accumulated elapsed time consumed executing user-defined functions on the main application execution unit. Since these UDFs run entirely within DB2, this time represents class 1 and class 2 time.

**Field Name:** QWACUDFNF_ELAP

**CLASS 1/2 UDF NF CP CPU TIME**

Accumulated CPU time consumed executing user-defined functions on the main application execution unit. This time does not include CPU consumed on an IBM specialty engine. Since these UDFs run entirely within DB2, this time represents class 1 and class 2 time.

**Field Name:** QWACUDFNF_CP

**CLASS 2: DB2 ELAPSED TIME**

The class 2 elapsed time for nonnested activity accumulated in DB2 for the allied agent. This time does not include the time spent in DB2 processing SQL statements issued by stored procedures, user-defined functions, or triggers.

**Special Considerations**
• The time for most thread allocation and certain abend conditions is not reflected in this time.
• The elapsed time for distributed processing is included in the elapsed time of allied-distributed threads.
• In query CP, sysplex query, or utility parallelism, this is the time shown in the originating record, which overlaps the elapsed times shown in the parallel records.

**Note:** This field is not normally shown in the short layouts but can be included with UTR.

**Field Name:** QWACASC

**CLASS 2: DB2 ENTRY/EXIT EVENTS**

The total number of DB2 entry and exit events processed by the allied address space to calculate the elapsed time in DB2 and the processor time. This counter does not include the SQL entry and exit events processed by stored procedures.

**Field Name:** QWACARNA

This is an exception field.

**CLASS 2: TCB TIME**

The accumulated MVS CPU time that is spent in DB2. This CPU time does not include the:
• CPU time that is consumed on an IBM specialty engine
• CPU time that is consumed while processing SQL statements in a stored procedure

**Field Name:** QWCAJST

**CLASS 2: NON-ZERO CLASS 2**

This data section shows whether there is nonzero accounting class 2 data. Yes indicates that accounting class 2 or 7 was active during the life of the agent when a class 2 event occurred.
FIELD NAME: QWACCLS2
CLASS 2: STORED PROC ELAPSED TIME
The total elapsed time that the allied agent spent executing SQL in the stored procedures or WLM address space.
A stored procedure may initiate a trigger or invoke a user-defined function this time is not included in this counter.

Note: This field is not normally shown in the short layouts but can be included with UTR.

FIELD NAME: QWACSPNE
CLASS 2: UDF ELAPSED TIME
The total elapsed time spent by the allied agent executing SQL using UDF requests processed in a DB2 stored procedure or WLM address space. A user-defined function may initiate a trigger or invoke a stored procedure. Any time spent there is not included in this counter. This time includes time needed to connect and disconnect the UDF task. Non-inline UDF times are included in this time if the native UDF was called on a nested task and was not invoked by the main application execution unit.

Note: With user-tailored reporting (UTR) you can include this field in the short layouts of Accounting.

FIELD NAME: QWACUDEB
CLASS 2: UDF SQL ENTRY/EXITS EVENTS
The number of SQL entry/exit events performed by user-defined functions. This is only calculated if accounting class 2 is active.

FIELD NAME: QWACCL2O
CLASS 2: STORED PROCEDURE TCB TIME
The TCB time accumulated in DB2 for processing SQL statements issued by stored procedures. This time is only calculated if accounting class 2 is active.

FIELD NAME: QWACPTT
CLASS 2: STORED PROC. ENTRY/EXITS
The number of SQL entry or exit events performed by stored procedures. This number is only calculated if accounting class 2 is active.

FIELD NAME: QWACDEB
CLASS 2: CP CPU TIME UDF
The accumulated CPU time consumed in DB2 when processing SQL statements that were issued by UDF(s) processed in a DB2 stored procedure or WLM address space.
This time also includes the DB2 time required to connect and disconnect the UDF task. Non-inline UDF times are included in this time if the native UDF was called on a nested task and was not invoked by the main application execution unit.

This time is a subset of QWACUDCP and is only calculated if accounting class 2 is active.

This time does not include CPU consumed on an IBM specialty engine.

**Field Name:** QWACUDTT

**CLASS 2: SE CPU TIME**

The accumulated and consumed class 2 time on an IBM specialty engine (SE) that consists of times for non-nested, stored procedures, user-defined functions, triggers, and parallel tasks.

**Note:** All CPU times of an IBM specialty engine that are reported in DB2 trace records are already normalized by DB2 to the speed of the purpose processor.

**Field Name:** QWACCLS2_ZIIP

**CLASS 2: TRIG ELAP TIME UNDER ENCLAVE**

The accumulated elapsed time used for executing triggers under an enclave.

**Field Name:** QWACTREE

**CLASS 2: SE ELIGIBLE CP CPU TIME**

The accumulated CPU time that ran on a standard CP for work eligible on an IBM specialty engine.

For records for the parent tasks in parallel queries, this value reflects zIIP-eligible time for the parent and the child tasks. Child task records have a value of 0.

**Field Name:** QWACZIIP_ELIGIBLE

**CLASS 2: TRIG TCB TIME UNDER ENCLAVE**

The accumulated CPU time used for executing triggers on a nested task. This time does not include CPU consumed on an IBM specialty engine.

**Field Name:** QWACTRTE

**CLASS 2: QWACTRTRT_ZIIP**

The accumulated CPU time consumed on an IBM specialty engine while running triggers on a nested task or on the main application execution unit.

**Field Name:** QWACTRTRT_ZIIP

**CLASS 2: TRIG ELAP TIME NOT UNDER ENCLAVE**

The accumulated elapsed time used when executing under the control of a trigger. This does not include the time used while in user-defined functions or stored procedures that are called from the trigger.

**Field Name:** QWACTRET

**CLASS 2: TRIG TCB TIME NOT UNDER ENCLAVE**
The accumulated TCB time that is used when running under the control of a trigger. This does not include the time that is used while running in user-defined functions or stored procedures that are called from the trigger.

This CPU time does not include the CPU time that is consumed on an IBM specialty engine.

**Field Name:** QWACTRTT

This is an *exception* field.

### CLASS 3: ACCU LOCK AND LATCH ELAPSED TIME (DB2 9)

This field depends on the DB2 version that is installed:

- **DB2 10 or later:** The accumulated wait time because of local contention for locks. The term *local contention* is used to differentiate from *global contention* (which is reported in QWACAWTJ). Local contention does not require intersystem communication. The contention is detected and resolved entirely within this subsystem.

- **Prior to DB2 10:** The accumulated lock and latch elapsed time. It indicates the elapsed time the allied agent waited for locks and latches in DB2. This value does not include suspensions because of group-level lock contentions in a data sharing environment. When the event completes, the ending time is used to calculate the total elapsed wait time. The result is added to the previously saved lock and latch wait time in DB2. If class 9 is not active, the time shown does not include the time for internal latching.

**Field Name:** QWACAWTL

This is an *exception* field.

### CLASS 3: WAIT TRACE EVENTS (Prior to DB2 10)

This field depends on the DB2 version that is installed:

- **DB2 10 or later:** The number of wait trace events processed for waits for local contention for locks.

- **Prior to DB2 10:** The number of wait trace events processed for lock/latch.

**Field Name:** QWACARNL

### CLASS 3: ACCU LOCK ELAPSED TIME (DB2 10 or later)

This field depends on the DB2 version that is installed:

- **DB2 10 or later:** The accumulated wait time because of local contention for locks. The term *local contention* is used to differentiate from *global contention* (which is reported in QWACAWTJ). Local contention does not require intersystem communication. The contention is detected and resolved entirely within this subsystem.

- **Prior to DB2 10:** The accumulated lock and latch elapsed time. It indicates the elapsed time the allied agent waited for locks and latches in DB2. This value does not include suspensions because of group-level lock contentions in a data sharing environment. When the event completes, the ending time is used to calculate the total elapsed wait time. The result is added to the previously saved lock and latch wait time in DB2. If class 9 is not active, the time shown does not include the time for internal latching.

**Field Name:** QWACAWTL
This is an exception field.

**CLASS 3: WAIT TRACE EVENTS (DB2 10 or later)**

This field depends on the DB2 version that is installed:
- **DB2 10 or later**: The number of wait trace events processed for waits for local contention for locks.
- **Prior to DB2 10**: The number of wait trace events processed for lock/latch.

**Field Name:** QWACARNL

**CLASS 3: DB2 LATCH SUSP TIME (DB2 10 or later)**

The accumulated wait time because of latch contention.

**Field Name:** QWACAWLH

**CLASS 3: LATCH WAIT TRACE EVENTS (DB2 10 or later)**

The number of wait trace events processed for waits for latch contention.

**Field Name:** QWACARLH

**CLASS 3: SYNCHRONOUS I/O SUSP TIME**

The accumulated I/O elapsed wait time for database I/O done under this thread. This field is for synchronous I/O only. It includes synchronous read and write I/O. This value is an average.

**Field Name:** QWACAWTI

**CLASS 3: SYNCHRONOUS I/O SUSP EVENTS**

The number of wait trace events processed for I/O.

**Field Name:** QWACARNE

**CLASS 3: LOG WRITE I/O SUSP TIME**

The accumulated wait time for log write I/O.

This value is an average.

**Field Name:** QWACAWLG

**CLASS 3: LOG WRITE I/O SUSP EVENTS**

The number of log I/O suspensions.

**Field Name:** QWACARLG

**CLASS 3: OTHER READ SUSP TIME**

The accumulated waiting time due to a read I/O that performed under a thread other than the one being reported. The time does not represent the total duration of the subject read I/O. It includes:
- Sequential prefetch
- List prefetch
- Sequential detection
- Synchronous read I/O performed by a thread other than the one being reported

**Field Name:** QWACAWTR

This is an exception field.

**CLASS 3: OTHER READ SUSP EVENTS**
The number of suspensions due to read I/O.

**Field Name:** QWACARNR

**CLASS 3: OTHER WRITE SUSP TIME**

The accumulated waiting time due to a write I/O that performed under a thread other than the one being reported. This time does not represent the total duration of the subject write I/O. It includes:

- An asynchronous write I/O
- A synchronous write I/O performed by a thread other than the one being reported

**Field Name:** QWACAWTW

This is an exception field.

**CLASS 3: OTHER WRITE SUSP EVENTS**

The number of suspensions due to write I/O.

**Field Name:** QWACARNW

**CLASS 3: UPDATE COMMIT SUSP TIME**

The accumulated wait time because of synchronous execution unit switch for DB2 Phase 2 commit, abort, or deallocation. This includes wait time for Phase 2 commit Log writes and database writes for LOB with LOG NO. For data sharing environment Page P-locks unlocks for updated pages and GBP writes.

**Field Name:** QWACAWTE

**CLASS 3: UPDATE COMMIT SUSP EVENTS**

The number of update commit suspensions.

**Field Name:** QWACARNS

**CLASS 3: PAGE LATCH (DB2+IRLM) SUSP TIME**

In the data sharing environment, within the same member, the first thread gets a P-lock (such as: Index leaf page P-Lock or P-Lock for Space map page or data page P-lock for Row level locking). With a high number of concurrent threads, for subsequent threads in the same member for the same resource, contention is reported as encountering a page latch contention. Randomizing the Index key helps minimizing page latch contentions for the Index leaf page. The Member Cluster option reduces page latch contention for a Space map page.

**Field Name:** QWACAWTP

**CLASS 3: PAGE LATCH (DB2+IRLM) SUSP EVENTS**

The number of page latch wait trace events processed.

**Field Name:** QWACARNH

**CLASS 3: NOTIFY MESSAGES SUSP TIME**

The accumulated elapsed waiting time due to suspensions caused by sending notify messages to other members in the data sharing group. Messages are sent, for example, when the database descriptors are changed due to DDL.

**Field Name:** QWACAWTG
CLASS 3: NOTIFY MESSAGES EVENTS
The number of wait trace events processed for sending notify messages to other members in the data sharing group.

Field Name: QWACARNG

CLASS 3: GLOB CONT PARENT L-LOCK TIME
The accumulated global contention wait time for parent L-locks.
A parent L-lock can be one of the following types:
- Database
- Tablespace
- Table
- Partition

Background and Tuning Information
Performance Expert might adjust this value if the thread was suspended when performance data was gathered.

Field Name: QWACAWTJ

CLASS 3: GLOB CONT PARENT L-LOCK EVENTS
The number of wait trace events processed for group-level contentions in a data sharing environment.

Field Name: QWACARNJ

CLASS 3: GLOB CONT CHILD L-LOCK TIME
The accumulated global contention wait time for child L-locks.
A child L-lock type can be:
- Page
- Row

Field Name: QWACAWTK

CLASS 3: GLOB CONT CHILD L-LOCK EVENTS
The number of wait trace events processed for waits due to global contention for child L-locks.

Field Name: QWACARNK

CLASS 3: GLOB CONT OTHER L-LOCK TIME
The accumulated global contention wait time for other L-locks. Global extend lock is acquired in exclusive mode by Inserters before an extend service task switch.

Field Name: QWACAWTM

CLASS 3: GLOB CONT OTHER L-LOCK EVENTS
The number of wait trace events processed for waits due to global contention for other L-locks.

Field Name: QWACARNM

CLASS 3: GLOB CONT PGSET/PART P-LOCK TIME
The accumulated global contention time for pageset and partition P-locks.

Field Name: QWACAWTN
CLASS 3: GLOB CONT PGSET/PART P-LOCK EVENTS
The number of wait trace events processed for waits due to global contention for page set or partition P-locks.
Field Name: QWACARNN

CLASS 3: GLOB CONT PAGE P-LOCK TIME
The accumulated global contention wait time for page P-locks.
Field Name: QWACAWTO

CLASS 3: GLOB CONT PAGE P-LOCK EVENTS
The number of wait trace events processed for waits due to global contention for page P-locks.
Field Name: QWACARNO

CLASS 3: GLOB CONT OTHER P-LOCK TIME
The accumulated global contention wait time for other P-locks. Includes suspension for Castout P-Locks and DBET locks. It could be because of Index Split processing which can be minimized if the Index key size is not large. If you can minimize the number of Index Keys in the Index, it will help to reduce the number of Index splits. For DB2 9, a large Index page size could also reduce the Index splits.
Field Name: QWACAWTQ

CLASS 3: GLOB CONT OTHER P-LOCK EVENTS
The number of wait trace events processed for waits due to global contention for other P-locks.
Field Name: QWACARNQ

CLASS 3: SCHED. STOR PROC SUSP TIME
The total elapsed waiting time for an available TCB before the stored procedure could be scheduled.
Field Name: QWACCAST

CLASS 3: STORED PROCEDURE EVENTS
The number of wait trace events processed for an unavailable TCB needed for a stored procedure.
Field Name: QWACCANM

CLASS 3: SCHED. UDF SUSP TIME
The total elapsed time spent waiting for an available TCB before the user-defined function could be scheduled.
Field Name: QWACUDST

CLASS 3: NON-ZERO CLASS 3
CLASS 3 DATA IN THIS RECORD:Y/N
Field Name: QWACCLS3

CLASS 3: TCP/IP LOB XML TIME
The accumulated wait time for TCP/IP LOB and XML (storing large object and XML) materialization.
Field Name: QWACALBW
IFCID 003 - Instrumentation Accounting Data

CLASS 3: TCP/IP LOB XML EVENTS
The number of wait trace events that were processed for waits for TCP/IP LOB and XML materialization.
Field Name: QWACALBC

CLASS 3: ACCELERATOR SUSP TIME
The accumulated wait time for requests to an accelerator.
Field Name: QWACAACW

CLASS 3: ACCELERATOR EVENTS
The number of wait trace events processed for requests to an accelerator.
Field Name: QWACAACC

CLASS 3: CLASS 3 DATA COLLECTED
The accounting class 3 data was being collected when this accounting record was written.
Field Name: QWACCL3O

CLASS 7: DATA COLLECTED
The accounting class 7 data was being collected when this accounting record was written.
Field Name: QWACCL7O

CLASS 8: DATA COLLECTED
The accounting class 8 data was being collected when this accounting record was written.
Field Name: QWACCL8O

WLM SERVICE CLASS
The MVS workload manager service class name. This field is used for database access threads on MVS 5.2 or later.
Field Name: QWACWLME

PARALLEL CHILDS ROLLED INTO RECORD
The number of parallel child agents rolled into this record. The value depends on the record type:
1. For all non-rollup records, this value is 0.
2. For a parallel query rollup record, this value is the number of parallel child agents rolled into this record.
3. For a DDF/RRSAF rollup record, this value is the number of parallel query child agents rolled into this record. These agents are NOT counted in QWACPCNT.
4. For an autonomous procedure rollup record, this value is 0.
Field Name: QWAC_PT_COUNT

ROLLUP DATA FOR PARALLEL CHILD TASKS
The field indicates whether to roll up accumulate query parallel task's accounting trace into originating task's accounting trace. Possible values are:
YES  Originating task cut an additional accounting trace record with all roll-up values from parallel tasks.

NO  Each parallel task will produce its own accounting trace record.

Field Name: QWACPARR
IFCID 003 - Instrumentation Accounting Data Overflow

**IFCID 003 - Instrumentation Accounting Data Overflow**

This topic shows detailed information about “Record Trace - IFCID 003 - Instrumentation Accounting Data Overflow”.

**Record trace - IFCID 003 - Instrumentation Accounting Data Overflow**

The field labels shown in the following sample layout of “Record Trace - IFCID 003 - Instrumentation Accounting Data Overflow” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARCH.LOG(QUIES) SUSP TIME</td>
<td>The accumulated waiting time due to the processing of ARCHIVE LOG MODE(QUIESCE) commands. This time does not represent the time required to perform the entire command. Field Name: QWAXALOG This is an exception field.</td>
</tr>
<tr>
<td>DRAIN LOCK SUSP TIME</td>
<td>The accumulated waiting time for a drain lock. This is the time the requester is suspended while waiting to acquire the drain lock. Field Name: QWAXAWDR This is an exception field.</td>
</tr>
<tr>
<td>CLAIM RELEASE SUSP TIME</td>
<td>The accumulated waiting time for a drain waiting for claims to be released. After the drain lock is acquired, the drainer must wait for claim holders to release the object. Field Name: QWAXAWCL This is an exception field.</td>
</tr>
<tr>
<td>SYSLOGRNG SUSP TIME</td>
<td>The accumulated waiting time due to the processing of DRAIN LOG commands. This time does not represent the time required to perform the entire command. Field Name: QWAXARND</td>
</tr>
<tr>
<td>OTHER SERVICE SUSP TIME</td>
<td>The accumulated waiting time due to the processing of other service suspensions. This time does not represent the time required to perform the entire command. Field Name: QWAXAOSR</td>
</tr>
<tr>
<td>COMMIT PH1 WRITE I/O TIME</td>
<td>The accumulated waiting time due to the processing of COMMIT PH1 WRITE I/O commands. This time does not represent the time required to perform the entire command. Field Name: QWAXACMP</td>
</tr>
<tr>
<td>ASYNCH. IXL REQ. TIME</td>
<td>The accumulated waiting time for asynchronous IXL requests. This time does not represent the time required to perform the entire request. Field Name: QWAXAIXL</td>
</tr>
</tbody>
</table>

**ARCH.LOG(QUIES) SUSP TIME**

The accumulated waiting time due to the processing of ARCHIVE LOG MODE(QUIESCE) commands.

This time does not represent the time required to perform the entire command.

Field Name: QWAXALOG

This is an exception field.

**ARCH.LOG(QUIES) SUSP EVENTS**

The number of ARCHIVE LOG MODE (QUIESCE) commands issued.

Field Name: QWAXALCT

**DRAIN LOCK SUSP TIME**

The accumulated waiting time for a drain lock. This is the time the requester is suspended while waiting to acquire the drain lock.

Field Name: QWAXAWDR

This is an exception field.

**DRAIN LOCK SUSP EVENTS**

The number of wait trace events processed for waits for drain locks.

Field Name: QWAXARND

**CLAIM RELEASE SUSP TIME**

The accumulated waiting time for a drain waiting for claims to be released. After the drain lock is acquired, the drainer must wait for claim holders to release the object.

Field Name: QWAXAWCL

This is an exception field.

**CLAIM RELEASE SUSP EVENTS**

The number of wait trace events processed for waits for claims to be released.
IFCID 003 - Instrumentation Accounting Data Overflow

Field Name: QWAXARNC

OPEN/CLOSE SUSP TIME
Accumulated waiting time for a synchronous execution unit switch to the DB2 OPEN/CLOSE data set service for the HSM recall service.
This value is an average.

Field Name: QWAXOCSE

OPEN/CLOSE SUSP EVENTS
Number of wait trace events processed of waits for synchronous execution unit switching to the Open/Close service.

Field Name: QWAXOCNS

SYSLGRNG SUSP TIME
Accumulated wait time for a synchronous execution unit switch to the DB2 SYSLGRNG recording service. This service is sometimes used for Level ID checking for downlevel detection.
This value is an average.

Field Name: QWAXSLSE

SYSLGRNG SUSP EVENTS
Number of wait trace events for a synchronous execution unit switch to the DB2 SYSLGRNG recording service.

Field Name: QWAXSLNS

EXC/DEL/DEF SUSP TIME
Accumulated wait time for a synchronous execution unit switch to the DB2 data space manager services. This includes DEFINE DATA SET, EXTEND DATA SET, DELETE DATA SET, RESET DATA SET, and VSAM CATALOG ACCESS.
This value is an average.

Field Name: QWAXDSSE

EXC/DEL/DEF SUSP EVENTS
Number of wait trace events for waits for synchronous execution unit switching to the DB2 data space manager services.

Field Name: QWAXDSNS

OTHER SERVICE SUSP TIME
The VSAM catalog update. In the distributed environment, it includes the waiting time for the response from the server system.

Field Name: QWAXOTSE

OTHER SERVICE SUSP EVENTS
Number of wait trace events for a synchronous execution unit switch to other DB2 service tasks.

Field Name: QWAXOTNS

FORCE-AT-COMMIT SUSP TIME

Chapter 40. IFCID Record Blocks 40-191
IFCID 003 - Instrumentation Accounting Data Overflow

The accumulated time waiting for phase 1 commit write I/O. An example for this suspension is LOB Table Space with LOG NO Phase 1 commit database synchronous write I/O processing.

**Field Name:** QWAXAWFC

**FORCE-AT-COMMIT SUSP EVENTS**

The number of wait trace events for force-at-commit.

**Field Name:** QWAXFCCT

**ASYNC. IXL REQ TIME**

The accumulated wait time for IXLCACHE and IXLFCOMP requests.

**Field Name:** QWAXIXLT

**ASYNC. IXL EVENTS**

Number of wait trace events processed for asynchronous IXLCACHE or IXLFCOMP invocations.

**Field Name:** QWAXIXLE
IFCID 003 - Locking Data

This topic shows detailed information about “Record Trace - IFCID 003 - Locking Data”.

Record trace - IFCID 003 - Locking Data

The field labels shown in the following sample layout of “Record Trace - IFCID 003 - Locking Data” are described in the following section.

<table>
<thead>
<tr>
<th>LOCKING DATA</th>
<th>DEADLOCKS</th>
<th>0</th>
<th>LOCK REQUEST</th>
<th>39</th>
<th>LOCK SUSPENSIONS</th>
<th>0</th>
<th>CLAIM REQUESTS</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TIMEOUTS</td>
<td>0</td>
<td>UNLOCK REQUEST</td>
<td>31</td>
<td>ILRM LATCH SUSPENS.</td>
<td>1</td>
<td>CLAIM REQ. FAILED</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>ESCALATIONS(SHR)</td>
<td>0</td>
<td>QUERY REQUEST</td>
<td>0</td>
<td>OTHER SUSPENSIONS</td>
<td>0</td>
<td>DRAIN REQUESTS</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>ESCALATIONS(EXC)</td>
<td>0</td>
<td>CHANGE REQUEST</td>
<td>0</td>
<td>DRAIN REQ. FAILED</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MAXIMUM PAGE/ROW LOCKS HELD</td>
<td>1</td>
<td>OTHER REQUEST</td>
<td>0</td>
<td>OTHER REQUEST</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

DEADLOCKS

The number of times deadlocks were detected. This number should be low, ideally 0.

Background and Tuning Information

Deadlocks occur when two or more application processes each hold locks on resources that the others need, without which they cannot proceed. Ensure that all applications accessing the same tables access them in the same order.

Deadlocks can also occur through index page splits if there is high insert activity. In this case, the recommendation is to set SUBPAGES to 1 for the index.

This field is incremented once for each deadlock encountered. There is no correlation between this field and the deadlock events reported in the Locking report set or the number of IFCID 172 records written. This field reports all deadlocks, regardless of how they were resolved. The locking report and record trace IFCID 172 show only those deadlocks that were resolved by DB2.

Field Name: QTXADEA

This is an exception field.

LOCK REQUEST

The number of requests to lock a resource.

Field Name: QTXALOCK

This is an exception field.

LOCK SUSPENSIONS

The number of times a lock could not be obtained and the unit of work was suspended.

Background and Tuning Information

This number should be low, ideally 0.

The number of lock suspensions is a function of the lock requests. Lock suspensions (or conflicts) can happen on either LOCK REQUEST or CHANGE REQUEST.

Suspensions are highly dependent on the application and table space locking protocols.

Field Name: QTXASLOC
This is an exception field.

CLAIM REQUESTS
The number of claim requests.
Field Name: QTXACLNO
This is an exception field.

TIMEOUTS
The number of times a unit of work was suspended for a time exceeding the timeout value. This number should be low, ideally 0.
Field Name: QTXATIM
This is an exception field.

UNLOCK REQUEST
The number of requests to unlock a resource.
This value can be less than the number of lock requests because DB2 can release several locks with a single unlock request.
Field Name: QTXAUNLK

IRLM LATCH SUSPENS.
The number of latch suspensions.
Field Name: QTXASLAT

CLAIM REQ. FAILED
The number of unsuccessful claim requests.
Field Name: QTXACLUN

ESCALATIONS(SHR)
The number of times the maximum page locks per table space are exceeded, and the table space lock escalates from a page lock (IS) to a table space lock (S) for this thread. You can specify the number of locks allowed per table space with the LOCKS PER TABLE(SPACE) parameter on the DB2 install panel DSNTPJ.

Background and Tuning Information
Escalations can cause unpredictable response times. Lock escalations should only happen when an application process updates or references (if repeatable read is used) more pages than normal.
Field Name: QTXALES
This is an exception field.

QUERY REQUEST
The number of query requests.
Field Name: QTXAQRY
This is an exception field.

OTHER SUSPENSIONS
The number of suspensions caused by something other than lock or latch.
Field Name: QTXASOTH
This is an exception field.

DRAIN REQUESTS
The number of drain requests.

Field Name: QTXADRNO
This is an exception field.

ESCALATIONS(EXC)
The number of times the maximum page locks per table space are exceeded and the table space lock escalates from a page lock (IX) to a table space lock (X).

Background and Tuning Information
Escalations can cause unpredictable response times. Lock escalations should only happen when an application process updates or references (if repeatable read is used) more pages than it normally does.

A useful rule of thumb is to compare the number of escalations (shared and exclusive) to the successful escalations (those that did not cause deadlocks and timeouts). If this value, or the number Lock escalations - shared and if the number of timeouts or deadlocks is also not 0, the timeout or deadlock is probably caused by the escalation.

If many escalations cause deadlocks and timeouts, the recommendation is to change the escalation threshold value. Use of ANY is extremely useful to prevent unnecessary and expensive page locks, for example locking all pages in a tablespace.

Lock escalations, shared or exclusive, should not be expected in a transaction environment.

Field Name: QTXALEX
This is an exception field.

CHANGE REQUEST
The number of change requests.

Field Name: QTXACHG
This is an exception field.

DRAIN REQ. FAILED
The number of unsuccessful drain requests.

Field Name: QTXADRUN
This is an exception field.

MAXIMUM PAGE/ROW LOCKS HELD
The maximum number of page or row locks concurrently held against all table spaces by a single application during its execution. This count is a high-water mark. It cannot exceed the LOCKS PER USER parameter on panel DSNTPFJ.

Field Name: QTXANPL
This is an exception field.

OTHER REQUEST
IFCID 003 - Locking Data

The number of requests to IRLM to perform a function other than LOCK, UNLOCK, QUERY, or CHANGE.

Field Name: QTXAIRLM

This is an exception field.
IFCID 003 - Logging

This topic shows detailed information about “Record Trace - IFCID 003 - Logging”.

Record trace - IFCID 003 - Logging

The field labels shown in the following sample layout of “Record Trace - IFCID 003 - Logging” are described in the following section.

<table>
<thead>
<tr>
<th>LOGGING</th>
<th>NUMBER OF LOG RECORDS WRITTEN</th>
<th>0</th>
<th>TOTAL BYTES WRITTEN</th>
<th>X'000000000000'</th>
</tr>
</thead>
</table>

**NUMBER OF LOG RECORDS WRITTEN**

The number of log records written.

**Field Name:** QWACLRN

**TOTAL BYTES WRITTEN**

The total number of log record bytes written.

**Field Name:** QWACLRAB
IFCID 003 - Miscellaneous

IFCID 003 - Miscellaneous
This topic shows detailed information about “Record Trace - IFCID 003 - Miscellaneous”.

This report has the same layout as “IFCID 002 - Miscellaneous” on page 40-116.
IFCID 003 - Nested SQL Activity

This topic shows detailed information about “Record Trace - IFCID 003 - Nested SQL Activity”.

Record trace - IFCID 003 - Nested SQL Activity

The field labels shown in the following sample layout of “Record Trace - IFCID 003 - Nested SQL Activity” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAX CASCAD LVL</td>
<td>The maximum level of indirect SQL cascading. This includes cascading because of triggers, UDFs, or stored procedures.</td>
</tr>
<tr>
<td>CALL STATEMENTS</td>
<td>The number of SQL CALL statements executed.</td>
</tr>
<tr>
<td>PROCEDURE ABENDS</td>
<td>The number of times a stored procedure terminated abnormally.</td>
</tr>
<tr>
<td>CALL TIMEOUTS</td>
<td>The number of times an SQL call timed out waiting to be scheduled.</td>
</tr>
<tr>
<td>CALL REJECTS</td>
<td>The number of times an SQL CALL statement was rejected due to the procedure being in the STOP ACTION(REJECT) state.</td>
</tr>
<tr>
<td>UDF EXECUTED</td>
<td>The number of user-defined functions executed.</td>
</tr>
<tr>
<td>UDF ABENDS</td>
<td>The number of times a user-defined function abended.</td>
</tr>
</tbody>
</table>

Field Name: QXCASCADP
This is an exception field.

Field Name: QXCALL
This is an exception field.

Field Name: QXCALLAB
This is an exception field.

Field Name: QXCALLTO
This is an exception field.

Field Name: QXCALLRJ
This is an exception field.

Field Name: QXCAUD
This is an exception field.

Field Name: QXCAUDAB
This is an exception field.
IFCID 003 - Nested SQL Activity

**UDF TIMEOUTS**

The number of times a user-defined function timed out while waiting to be scheduled.

**Field Name:** QXCAUDTO

This is an *exception* field.

**UDF REJECTS**

The number of times a user-defined function was rejected.

**Field Name:** QXCAUDRJ

This is an *exception* field.

**STMT TRIGGER**

The number of times a statement trigger was activated.

**Field Name:** QXSTTRG

This is an *exception* field.

**ROW TRIGGER**

The number of times a row trigger was activated.

**Field Name:** QXROWTRG

This is an *exception* field.

**SQL ERROR TRIGGER**

The number of times an SQL error occurred during the execution of a triggered action. This includes errors that occur in user-defined functions or stored procedures that are called from triggers and that pass back a negative SQLCODE.

**Field Name:** QXTRGERR

This is an *exception* field.
IFCID 003 - Query Parallelism

This topic shows detailed information about “Record Trace - IFCID 003 - Query Parallelism”.

This report has the same layout as “IFCID 002 - Query Parallelism” on page 40-120.
IFCID 003 - Resource Limit Facility

IFCID 003 - Resource Limit Facility

This topic shows detailed information about “Record Trace - IFCID 003 - Resource Limit Facility”.

Record trace - IFCID 003 - Resource Limit Facility

The field labels shown in the following sample layout of “Record Trace - IFCID 003 - Resource Limit Facility” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RES LIMIT SCOPE</td>
<td>Indicates how the resource limit was established. A value of 0 shows that</td>
</tr>
<tr>
<td></td>
<td>the resource limit facility was not started.</td>
</tr>
<tr>
<td>QTXAPREC</td>
<td></td>
</tr>
<tr>
<td>RLF TABLE ID</td>
<td>The identifier of the resource limit specification table.</td>
</tr>
<tr>
<td>QTXARLID</td>
<td></td>
</tr>
<tr>
<td>LIMIT IN CPU</td>
<td>The CPU time limit, in microseconds, set by the resource limit facility.</td>
</tr>
<tr>
<td>16 MICROSEC</td>
<td></td>
</tr>
<tr>
<td>QTXACLMT</td>
<td></td>
</tr>
<tr>
<td>RES LIMIT TYPE</td>
<td>Indicates how the type of resource limit was established: infinite, zero,</td>
</tr>
<tr>
<td></td>
<td>or limit.</td>
</tr>
<tr>
<td>QTXAFLG1</td>
<td>Note: Label QTXAFLG1 presents the first flag byte in hexadecimal:</td>
</tr>
<tr>
<td></td>
<td>X'80'  Infinite limit</td>
</tr>
<tr>
<td></td>
<td>X'40'  No run or zero limit</td>
</tr>
<tr>
<td>LIMIT IN SERVICE</td>
<td>The maximum number of CPU service units to be used. Normally, the value</td>
</tr>
<tr>
<td>UNITS</td>
<td>is not 0 if the RES LIMIT TYPE is LIMIT. A value of 0 indicates no limit.</td>
</tr>
<tr>
<td>QTXASLMT</td>
<td></td>
</tr>
<tr>
<td>HIGHEST CPU</td>
<td>The highest CPU time used by a single DB2 call, in microseconds. Note</td>
</tr>
<tr>
<td>16 MICROSEC USED</td>
<td>that there can be many DB2 calls for one SQL statement.</td>
</tr>
<tr>
<td>QTXACHUS</td>
<td></td>
</tr>
<tr>
<td>QTXAFLG1 (S)</td>
<td>Indicates how the type of resource limit was established: infinite, zero,</td>
</tr>
<tr>
<td></td>
<td>or limit.</td>
</tr>
<tr>
<td>Note: Label QTXAFLG1</td>
<td>presents the first flag byte in hexadecimal:</td>
</tr>
<tr>
<td></td>
<td>X'80'  Infinite limit</td>
</tr>
</tbody>
</table>

RES LIMIT SCOPE

Indicates how the resource limit was established. A value of 0 shows that the resource limit facility was not started.

Field Name: QTXAPREC

RLF TABLE ID

The identifier of the resource limit specification table.

Field Name: QTXARLID

LIMIT IN CPU 16 MICROSEC

The CPU time limit, in microseconds, set by the resource limit facility.

Field Name: QTXACLMT

RES LIMIT TYPE

Indicates how the type of resource limit was established: infinite, zero, or limit.

Note: Label QTXAFLG1 presents the first flag byte in hexadecimal:

X'80'  Infinite limit
X'40'  No run or zero limit

Field Name: QTXAFLG1

LIMIT IN SERVICE UNITS

The maximum number of CPU service units to be used. Normally, the value is not 0 if the RES LIMIT TYPE is LIMIT. A value of 0 indicates no limit.

Field Name: QTXASLMT

HIGHEST CPU 16 MICROSEC USED

The highest CPU time used by a single DB2 call, in microseconds. Note that there can be many DB2 calls for one SQL statement.

Field Name: QTXACHUS

QTXAFLG1 (S)

Indicates how the type of resource limit was established: infinite, zero, or limit.

Note: Label QTXAFLG1 presents the first flag byte in hexadecimal:

X'80'  Infinite limit
X'40'   No run or zero limit

Field Name: QTXAFLG1
IFCID 003 - RID List Processing

IFCID 003 - RID List Processing
This topic shows detailed information about “Record Trace - IFCID 003 - RID List Processing”.

This report has the same layout as “IFCID 002 - RID List Processing” on page 40-124.
IFCID 003 - Rollup Accounting Correlation Block

IFCID 003 - Rollup Accounting Correlation Block (DB2 10 or later)

This topic shows detailed information about “Record Trace - IFCID 003 - Rollup Accounting Correlation Block (DB2 10 or later)”.

Record trace - IFCID 003 - Rollup Accounting Correlation Block (DB2 10 or later)

The field labels shown in the following sample layout of “Record Trace - IFCID 003 - Rollup Accounting Correlation Block (DB2 10 or later)” are described in the following section.

**TOKEN**

The agent token for the transaction rolled into the record. This can be used to correlate to records written with the same QWHSACE value during the time of the transaction (QWARBSC to QWARESC).

Field Name: QWARACE

**START**

The beginning time for the transaction.

Field Name: QWARBSC

**END**

The end time for the transaction.

Field Name: QWARESC
IFCID 003 - ROWID

IFCID 003 - ROWID
This topic shows detailed information about “Record Trace - IFCID 003 - ROWID”.

Record trace - IFCID 003 - ROWID
The field labels shown in the following sample layout of “Record Trace - IFCID 003 - ROWID” are described in the following section.

| ROWID | DIRECT ACCESS | 0 | INDEX USED | 0 | TABLE SPACE SCAN USED | 0 |

DIRECT ACCESS
The number of times that direct row access was successful.

Field Name: QXROIMAT

INDEX USED
The number of times an index was used to find a record.

Field Name: QXROIINX

TABLE SPACE SCAN USED
The number of times that an attempt to use direct row access reverted to using a table-space scan because DB2 was unable to use a matching index scan.

Background and Tuning Information
Ideally, this value should be 0.

Table-space scans can happen, for example, when a REORG is performed between the read of the ROWID column and the use of the host variable in the WHERE clause of the SQL statement. This causes the RID value in the host variable to be incorrect. DB2 first tries a matching-index scan before using a table-space scan.

To avoid table space scans, you can force the access path of an unsuccessful direct row access to use a matching index scan on the primary-index key by adding PKCOL to the WHERE clause in the SQL statement. .... WHERE ROWIDCOL=:HVROWID AND PKCOL=:HVPK

Field Name: QXROITS
IFCID 003 - SQL Call Data

This topic shows detailed information about “Record Trace - IFCID 003 - SQL Call Data”.

This report has the same layout as “IFCID 002 - SQL Call Data” on page 40-135.
IFCID 004 - Trace Start

This topic shows detailed information about “Record Trace - IFCID 004 - Trace Start”.

Record trace - IFCID 004 - Trace Start

The field labels shown in the following sample layout of “Record Trace - IFCID 004 - Trace Start” are described in the following section.

MESSAGE: −START TRACE (A )C (01 02 03 05 07 08 30 )
RMID (∗ )D (OPX )PLAN (∗ )AUTHID (∗ )IFCID
(3 )BUFSIZE (1024 )TDATA (COR CPU DIST )
QW0004CM X'200000D702001B02'

MESSAGE

The start trace message.

Field Name: QW0004MS
This topic shows detailed information about “Record Trace - IFCID 005 - Trace Stop”.

**Record trace - IFCID 005 - Trace Stop**

The field labels shown in the following sample layout of “Record Trace - IFCID 005 - Trace Stop” are described in the following section.

MESSAGE:  -STOP TRACE(*) CLASS(*) RMID(*) PLAN(*)
AUTHID(*) TNO(*)
QW0005CM  00000000

**MESSAGE**

The stop trace message.

**Field Name:** QW0005MS
IFCID 006 - Read I/O Start

This topic shows detailed information about “Record Trace - IFCID 006 - Read I/O Start”.

Record trace - IFCID 006 - Read I/O Start

The field labels shown in the following sample layout of “Record Trace - IFCID 006 - Read I/O Start” are described in the following section.

DBID: FIJ1DB01  POOL ID: 1  ACE : 1
OBID: FIJCCONT  FIRST: X’000002’  READTYPE: R
TABLE_SPACE_TYPE : N  PARTITION NUMBER: 0

DBID

The database ID. This is deduced from the DB2 fields QW0006DB, and QW0105DN or QW0107DN.

When present, the database name is shown, otherwise the decimal identifier from QW0006DB is shown, or N/A when this value is 0.

Field Name: RT0006DB

POOL ID

The internal identifier of the buffer pool. The values 0 through 49 are the identifiers for BP0 through BP49. The values 80 through 89 are the identifiers for BP32K through BP32K9.

Field Name: QW0006BP

ACE

The agent control element (ACE) token of the requester.

The relative number of the agent control element address in the ACE cross-reference table. This table is printed at the end of each location for every trace specified.

Field Name: QW0006AC

OBID

The object ID. This is deduced from the DB2 fields QW0006OB, and QW0105TN or QW0107TN.

When present, the name of the object is shown, otherwise the decimal identifier from QW0006OB is shown, or N/A when this value is 0.

Field Name: RT0006OB

FIRST

The hexadecimal number of the first page to be read for a table space that is not defined as large.

Field Name: QW0006PN

READTYPE

The type of read performed:

S      Sequential prefetch request
L      List prefetch request
D      Dynamic sequential prefetch request
IFCID 006 - Read I/O Start

R  Synchronous read request

Field Name: QW0006F

TABLE_SPACE_TYPE

The type of the table space:

L  Non-EA large table
N  Non-large table
V  EA-enabled large table

Field Name: QW0006FG

PARTITION_NUMBER

The partition number. This value is 0 if the table space is not partitioned.

Field Name: QW0006PT
Record trace - IFCID 007 - Read I/O Stop

The field labels shown in the following sample layout of “Record Trace - IFCID 007 - Read I/O Stop” are described in the following section.

<table>
<thead>
<tr>
<th>DBID</th>
<th>OBID</th>
<th>RETCODE</th>
<th>ACE</th>
<th>PAGE PREFETCHED VIA IO OPERATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSNDB01</td>
<td>SPT01</td>
<td>READ</td>
<td>12</td>
<td>X'000026F1' X'000026F2' X'000026F3' X'000026F4' X'000026F5' X'000026F6' X'000026F7' X'000026F8' X'000026F9' X'000026FA' X'000026FB' X'000026FC'</td>
</tr>
</tbody>
</table>

DBID

The database ID. This is deduced from the DB2 fields QW0007DB, QW0105DN or QW0107DN.

When present, the database name is shown, otherwise the decimal identifier from QW0007DB is shown, or N/A when this value is 0.

Field Name: RT0007DB

RETCODE

The return code from the media manager.

Field Name: QW0007MM

ACE

The relative number of the agent control element address in the ACE cross-reference table. This table is printed at the end of each location for every trace specified.

Field Name: QW0007AC

OBID

The object ID. This is deduced from the DB2 fields QW0007OB, QW0105TN or QW0107TN.

When present, the name of the object is shown, otherwise the decimal identifier from QW0007OB is shown, or N/A when this value is 0.

Field Name: RT0007OB

READ

The number of pages read.

Field Name: QW0007NP

PAGE PREFETCHED VIA IO OPERATION

The page was prefetched using an I/O operation.

Field Name: QW0007PF
IFCID 008 - Write I/O Synch

This topic shows detailed information about “Record Trace - IFCID 008 - Write I/O Synch”.

Record trace - IFCID 008 - Write I/O Synch

The field labels shown in the following sample layout of “Record Trace - IFCID 008 - Write I/O Synch” are described in the following section.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBID</td>
<td>DBHSR01</td>
</tr>
<tr>
<td>ACTIVE</td>
<td>218</td>
</tr>
<tr>
<td>OBID</td>
<td>HSRPDSFP</td>
</tr>
<tr>
<td>UPDATED</td>
<td>264</td>
</tr>
<tr>
<td>POOL ID</td>
<td>0</td>
</tr>
<tr>
<td>WRITTEN</td>
<td>1</td>
</tr>
<tr>
<td>WRITE TYPE</td>
<td>NORMAL</td>
</tr>
<tr>
<td>PAGE FAULTS</td>
<td>0</td>
</tr>
<tr>
<td>PARTITION NUMBER</td>
<td>0</td>
</tr>
</tbody>
</table>

DBID

The database ID. This is deduced from the DB2 fields QW0008DB, QW0105DN or QW0107DN.

When present, the database name is shown, otherwise the decimal identifier from QW0008DB is shown, or N/A when this value is 0.

Field Name: RT0008DB

ACTIVE

The number of active buffers in the pool.

Field Name: QW0008AB

OBID

The object ID. This is deduced from the DB2 fields QW0008OB, QW0105TN or QW0107TN.

When present, the name of the object is shown, otherwise the decimal identifier from QW0008OB is shown, or N/A when this value is 0.

Field Name: RT0008OB

UPDATED

The number of updated pages in the deferred write queue for the buffer pool that is identified in field QW0008BP.

Field Name: QW0008DW

POOL ID

The internal buffer pool identifier. The values 0 through 49 are the identifiers for BP0 through BP49. The values 80 through 89 are the identifiers for BP32K through BP32K9.

Field Name: QW0008BP

WRITTEN

The number of pages to be written.

Field Name: QW0008WR

WRITE TYPE

The type of write:

NORMAL Normal write to disk
IFCID 008 - Write I/O Synch

CASTOUT
Write to disk initiated by a castout from the coupling facility

Field Name: QW0008FC

PAGE FAULTS
The number of anticipated page faults. Real storage frames are tested before issuing write.

Field Name: QW0008PI

PARTITION NUMBER
The partition number. This value is 0 if the table space is not partitioned.

Field Name: QW0008PT
IFCID 009 - Write I/O

This topic shows detailed information about “Record Trace - IFCID 009 - Write I/O”.

Record trace - IFCID 009 - Write I/O

The field labels shown in the following sample layout of “Record Trace - IFCID 009 - Write I/O” are described in the following section.

RETURN

The return code from the media manager.

Field Name: QW009MM
IFCID 010 - Write I/O Asynch

Record trace - IFCID 010 - Write I/O Asynch

The field labels shown in the following sample layout of “Record Trace - IFCID 010 - Write I/O Asynch” are described in the following section.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBID</td>
<td>DBHSR01</td>
</tr>
<tr>
<td>OBID</td>
<td>HSRPDSFP</td>
</tr>
<tr>
<td>POOL ID</td>
<td>0</td>
</tr>
<tr>
<td>WRITTEN</td>
<td>1</td>
</tr>
<tr>
<td>WRITE TYPE</td>
<td>NORMAL</td>
</tr>
<tr>
<td>PAGE FAULTS</td>
<td>0</td>
</tr>
<tr>
<td>PARTITION NUMBER</td>
<td>0</td>
</tr>
</tbody>
</table>

**DBID**

The database ID. Deduced from the DB2 fields QW0010DB, and QW0105DN or QW0107DN.

When present the database name is shown, otherwise the decimal identifier from QW0010DB is shown, or N/A when this value is 0.

Field Name: RT0010DB

**ACTIVE**

The number of active buffers in the pool.

Field Name: QW0010AB

**OBID**

The object ID. Deduced from the DB2 fields QW0010OB, and QW0105TN or QW0107TN.

When present, the name of the object is shown, otherwise the decimal identifier from QW0010OB is shown or N/A when this value is 0.

Field Name: RT0010OB

**UPDATED**

The number of updated pages in the deferred write queue for the buffer pool that is identified in field QW0010BP.

Field Name: QW0010DW

**POOL ID**

The internal identifier of the buffer pool. The values 0 through 49 are the identifiers for BP0 through BP49. The values 80 through 89 are the identifiers for BP32K through BP32K9.

Field Name: QW0010BP

**WRITTEN**

The number of pages to be written.

Field Name: QW0010WR

**WRITE TYPE**

The type of write:

NORMAL

Normal write to disk
IFCID 010 - Write I/O Asynch

CASTOUT
  Write to disk initiated by a castout from the coupling facility

Field Name: QW0010FC

PAGE FAULTS
  The number of anticipated page faults. Real storage frames are tested before issuing write.

Field Name: QW0010PI

PARTITION NUMBER
  The partition number. This value is 0 if the table space is not partitioned.

Field Name: QW0010PT
IFCID 011 - Validate Exit

This topic shows detailed information about “Record Trace - IFCID 011 - Validate Exit”.

Record trace - IFCID 011 - Validate Exit

The field labels shown in the following sample layout of “Record Trace - IFCID 011 - Validate Exit” are described in the following section.

```
DBID 1  REC ID 3
OBID 2  TIME 4/01/08 17:42:00.000000
RETURN 0004  REASON 00000005
```

DBID

The database ID. Deduced from the DB2 fields QW0011DB, and QW0105DN or QW0107DN.

When present, the database name is shown, otherwise the decimal identifier from QW0011DB is shown, or N/A when this value is 0.

Field Name: RT0011DB

REC ID

The decimal identifier of the DB2 table OBID.

Field Name: QW0011OB

OBID

The object ID. Deduced from the DB2 fields QW0011OB, and QW0105TN or QW0107TN.

When present, the name of the object is shown, otherwise the decimal identifier from QW0011OB is shown or N/A when this value is 0.

Field Name: RT0011OB

TIME

The time at which the exit was called.

Field Name: QW0011TM

RETURN

The return code (EXPLRC1) from the exit.

Field Name: QW0011RT

REASON

The reason code (EXPLRC2) from the exit.

Field Name: QW0011RE
IFCID 012 - Edit Exit to Encode

This topic shows detailed information about “Record Trace - IFCID 012 - Edit Exit to Encode”.

Record trace - IFCID 012 - Edit Exit to Encode

The field labels shown in the following sample layout of “Record Trace - IFCID 012 - Edit Exit to Encode” are described in the following section.

<table>
<thead>
<tr>
<th>DBID</th>
<th>REC ID</th>
<th>TIME</th>
<th>RETURN</th>
<th>REASON</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>19</td>
<td>06/03/08</td>
<td>000A</td>
<td>00000014</td>
</tr>
</tbody>
</table>

DBID

The database ID. Deduced from the DB2 fields QW0012DB, QW0105DN or QW0107DN.

When present, the database name is shown, otherwise the decimal identifier from QW0012DB is shown, or N/A when this value is 0.

Field Name: RT0012DB

REC ID

The decimal identifier of the DB2 table OBID.

Field Name: QW0012OB

OBID

The object ID. Deduced from the DB2 fields QW0010OB, and QW0105TN or QW0107TN.

When present, the name of the object is shown, otherwise the decimal identifier from QW0010OB is shown or N/A when this value is 0.

Field Name: RT0010OB

TIME

The time at which the exit was called.

Field Name: QW0012TM

RETURN

The return code (EXPLRC1) from the exit.

Field Name: QW0012RT

REASON

The reason code (EXPLRC2) from the exit.

Field Name: QW0012RE
IFCID 013 - Hash Scan Input Start

This topic shows detailed information about “Record Trace - IFCID 013 - Hash Scan Input Start”.

Record trace - IFCID 013 - Hash Scan Input Start

The field labels shown in the following sample layout of “Record Trace - IFCID 013 - Hash Scan Input Start” are described in the following section.

<table>
<thead>
<tr>
<th>DBID</th>
<th>DSNDB01</th>
<th>REC ID</th>
<th>33</th>
</tr>
</thead>
<tbody>
<tr>
<td>OBID</td>
<td>DBD01</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COLUMN1</td>
<td>OPER</td>
<td>COL/VAL</td>
<td>CONN</td>
</tr>
<tr>
<td>10</td>
<td>NE</td>
<td>X'F2F4F54040404040'</td>
<td>A</td>
</tr>
<tr>
<td>15</td>
<td>GT</td>
<td>X'F2F4F54040404040'</td>
<td>A</td>
</tr>
</tbody>
</table>

DBID

The database ID. Deduced from the DB2 fields QW0010DB, and QW0105DN or QW0107DN.

When present the database name is shown, otherwise the decimal identifier from QW0010DB is shown, or N/A when this value is 0.

Field Name: RT0010DB

REC ID

The decimal identifier of the DB2 table OBID.

Field Name: QW0013OB

OBID

The object ID. Deduced from the DB2 fields QW0013OB, QW0105TN or QW0107TN.

When present, the name of the object is shown, otherwise the decimal identifier from QW0013OB is shown or N/A when this value is 0.

Field Name: RT0013OB

COLUMN1

The first column number.

Field Name: QW0013C1

OPER

The logical operator:

<table>
<thead>
<tr>
<th>NE</th>
<th>Not equal to</th>
</tr>
</thead>
<tbody>
<tr>
<td>GT</td>
<td>Greater than</td>
</tr>
<tr>
<td>GE</td>
<td>Greater than or equal to</td>
</tr>
<tr>
<td>LE</td>
<td>Less than or equal to</td>
</tr>
<tr>
<td>E</td>
<td>Equal</td>
</tr>
<tr>
<td>L</td>
<td>Less than</td>
</tr>
<tr>
<td>LT</td>
<td>Less than</td>
</tr>
<tr>
<td>LI</td>
<td>Like</td>
</tr>
<tr>
<td>NL</td>
<td>Not like</td>
</tr>
</tbody>
</table>
IFCID 013 - Hash Scan Input Start

?? Unknown operator
Field Name: QW0013OP

COL/VAL
Column data is printed in decimal. Value data is printed in hexadecimal.
Field Name: QW0013VA

CONN
The connector value:
A And
O Or
NONE Not specified
Field Name: QW0013CO

TRUE/FALSE
Indicates whether the comparison is true:
T True
F False
NONE Not specified
Field Name: QW0013TF
IFCID 014 - Hash Scan End

Record trace - IFCID 014 - Hash Scan End

The field labels shown in the following sample layout of “Record Trace - IFCID 014 - Hash Scan End” are described in the following section.

RETURN 0
QW0014RT 0
RETURN

The return code.

Field Name: QW0014RT
This topic shows detailed information about “Record Trace - IFCID 015 - Index Scan Begin”.

Record trace - IFCID 015 - Index Scan Begin

The field labels shown in the following sample layout of “Record Trace - IFCID 015 - Index Scan Begin” are described in the following section.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBID</td>
<td>The database ID. Deduced from the DB2 fields QW0015DB, QW0105DN or QW0107DN.</td>
</tr>
<tr>
<td>REC ID</td>
<td>The decimal identifier of the DB2 table OBID.</td>
</tr>
<tr>
<td>CUB</td>
<td>The hexadecimal address of the CUB token.</td>
</tr>
<tr>
<td>OBID</td>
<td>The object ID. Deduced from the DB2 fields QW0015OB, QW0105TN or QW0107TN.</td>
</tr>
<tr>
<td>INDX ID</td>
<td>The index identifier.</td>
</tr>
<tr>
<td>COLUMN1</td>
<td>The first column number.</td>
</tr>
<tr>
<td>OPER</td>
<td>The logical operator:</td>
</tr>
<tr>
<td>NE</td>
<td>Not equal to</td>
</tr>
<tr>
<td>GT</td>
<td>Greater than</td>
</tr>
<tr>
<td>GE</td>
<td>Greater than or equal to</td>
</tr>
</tbody>
</table>
IFCID 015 - Index Scan Begin

LE    Less than or equal to
E    Equal
L    Less than
LT    Less than
LI    Like
NL    Not like
??    Unknown operator

Field Name: QW0015OP

COL/VAL

Column data is printed in decimal. Value data is printed in hexadecimal.

Field Name: QW0015VA

CONN

The connector value:
A    And
O    Or
NONE    Not specified

Field Name: QW0015CO

TRUE/FALSE

Indicates whether the comparison is true:
T    True
F    False
NONE    Not specified

Field Name: QW0015TF
This topic shows detailed information about “Record Trace - IFCID 016 - Insert Scan Begin”.

**Record trace - IFCID 016 - Insert Scan Begin**

The field labels shown in the following sample layout of “Record Trace - IFCID 016 - Insert Scan Begin” are described in the following section.

<table>
<thead>
<tr>
<th>Field</th>
<th>DBID 260</th>
<th>REC ID 26</th>
<th>SQL TYPE</th>
<th>N/A</th>
<th>OBID 12</th>
<th>TRIGGER LEVEL</th>
<th>N/A</th>
<th>WORKFILE TYPE</th>
<th>N/A</th>
<th>CUB X'7F3F91EC'</th>
<th>INTEGRITY TYPE</th>
<th>N/A</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DBID</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| The database ID. Deduced from the DB2 fields QW0016DB, QW0105DN, or QW0107DN. When present, the database name is shown, otherwise the decimal identifier from QW0016DB is shown, or N/A when this value is 0. **Field Name:** RT0016DB
| **REC ID**  |          |           |          |      |         |               |      |               |      |                |                |      |
| The decimal identifier of the DB2 table OBID. **Field Name:** QW0016OB
| **SQL TYPE**|          |           |          |      |         |               |      |               |      |                |                |      |
| Possible values are: I INSERT | I INSERT | U UPDATE | Insert into a transition table for an UPDATE. D DELETE | Insert into a transition table for a DELETE. R RI | Insert into a transition table for a DELETE SET NULL for referential integrity. **Field Name:** QW0016ST
| **OBID**    |          |           |          |      |         |               |      |               |      |                |                |      |
| The object ID. Deduced from the DB2 fields QW0016OB, QW0105TN or QW0107TN. When present, the name of the object is shown, otherwise the decimal identifier from QW0016OB is shown, or N/A when this value is 0. **Field Name:** RT0016OB
| **TRIGGER LEVEL**|          |           |          |      |         |               |      |               |      |                |                |      |
| Depth of the trigger in the range 0 (no triggers) through 16. **Field Name:** QW0016TL
| **WORKFILE TYPE**|          |           |          |      |         |               |      |               |      |                |                |      |
| Possible values are: |         |           |          |      |         |               |      |               |      |                |                |      |
IFCID 016 - Insert Scan Begin

WF  Workfile
TT  Temporary Table
TR  Transition table
NW  Non-workfile

Field Name: QW0016WT

CUB

The hexadecimal address of the CUB token.

Field Name: QW0016AC

INTEGRITY TYPE

BLANK
S   SET NULL
   This can occur when SQL TYPE=U
C   CASCADE DELETE
   This can occur when SQL TYPE=D

Field Name: QW0016RI
IFCID 017 - Sequential Scan Begin

This topic shows detailed information about “Record Trace - IFCID 017 - Sequential Scan Begin”.

Record trace - IFCID 017 - Sequential Scan Begin

The field labels shown in the following sample layout of “Record Trace - IFCID 017 - Sequential Scan Begin” are described in the following section.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBID</td>
<td>The database ID. Deduced from the DB2 fields QW0017DB, QW0105DN or QW0107DN. When present, the database name is shown, otherwise the decimal identifier from QW0017DB is shown, or N/A when this value is 0. <strong>Field Name:</strong> RT0017DB</td>
</tr>
<tr>
<td>REC ID</td>
<td>The decimal identifier of the DB2 table OBID. <strong>Field Name:</strong> QW0017OB</td>
</tr>
<tr>
<td>CUB</td>
<td>The hexadecimal address of the CUB token. <strong>Field Name:</strong> QW0017AC</td>
</tr>
<tr>
<td>OBID</td>
<td>The object ID. Deduced from the DB2 fields QW0017OB, QW0105TN or QW0107TN. When present, the name of the object is shown, otherwise the decimal identifier from QW0017OB is shown, or N/A when this value is 0. <strong>Field Name:</strong> RT0017OB</td>
</tr>
<tr>
<td>SCAN TYPE</td>
<td>Possible values are: SQ Sequential scan. WF Work-file scan. TT Temporary table scan. TR Transition table scan for a trigger. <strong>Field Name:</strong> QW0017TY</td>
</tr>
<tr>
<td>COL1</td>
<td>The first column number. <strong>Field Name:</strong> QW0017C1</td>
</tr>
<tr>
<td>OP</td>
<td>5 E X'810E000000000000' 'NONE' 'NONE'</td>
</tr>
</tbody>
</table>
IFCID 017 - Sequential Scan Begin

The logical operator:

**NE**  Not equal to
**GT**  Greater than
**GE**  Greater than or equal to
**LE**  Less than or equal to
**E**  Equal
**L**  Less than
**LT**  Less than
**LI**  Like
**NL**  Not like
??  Unknown operator

Field Name: QW017OP

**COL/VAL**
Column data is printed in decimal. Value data is printed in hexadecimal.

Field Name: QW017VA

**CONN**
The connector type:

**A**  And
**O**  Or
**NONE**  Not specified

Field Name: QW017CO

**TRUE/FALSE**
Indicates whether the comparison is true:

**T**  True
**F**  False
**NONE**  Not specified

Field Name: QW017TF
IFCID 018 - Scan End

This topic shows detailed information about “Record Trace - IFCID 018 - Scan End”.

Record trace - IFCID 018 - Scan End

The field labels shown in the following sample layout of “Record Trace - IFCID 018 - Scan End” are described in the following section.

CUB X'7EBD2E60'
QW0018RT 4 QW0018RE 0

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATA TYPE INDX ROW PROC 5 ROW EXAM 5 STG1-QUAL 5 STG2-QUAL 0 ROW INSRT 0</td>
<td></td>
</tr>
<tr>
<td>CUB</td>
<td>The hexadecimal address of the cursor block token of the caller.</td>
</tr>
<tr>
<td>Field Name:</td>
<td>QW0018AC</td>
</tr>
<tr>
<td>DATA TYPE</td>
<td>The scan type identification:</td>
</tr>
<tr>
<td>INDX</td>
<td>Index scan</td>
</tr>
<tr>
<td>SEQD</td>
<td>Sequential data scan</td>
</tr>
<tr>
<td>SEQR</td>
<td>Transition table sequential data scan</td>
</tr>
<tr>
<td>SEQT</td>
<td>Temporary table sequential data scan</td>
</tr>
<tr>
<td>SEQW</td>
<td>Work-file sequential data scan</td>
</tr>
<tr>
<td>Field Name:</td>
<td>QW0018ID</td>
</tr>
<tr>
<td>ROW PROC</td>
<td>The number of rows processed.</td>
</tr>
<tr>
<td>Field Name:</td>
<td>QW0018RP</td>
</tr>
<tr>
<td>ROW EXAM</td>
<td>The number of rows examined. If DATA TYPE shows INDX, this number is the number of index entries (not rows) scanned.</td>
</tr>
<tr>
<td>Field Name:</td>
<td>QW0018LA</td>
</tr>
<tr>
<td>STG1-QUAL</td>
<td>The number of rows qualified at stage 1.</td>
</tr>
<tr>
<td>Field Name:</td>
<td>QW0018DQ</td>
</tr>
<tr>
<td>STG2-QUAL</td>
<td>The number of rows qualified at stage 2.</td>
</tr>
<tr>
<td>Field Name:</td>
<td>QW0018RQ</td>
</tr>
<tr>
<td>ROW INSRT</td>
<td>The number of rows inserted.</td>
</tr>
<tr>
<td>Field Name:</td>
<td>QW0018IN</td>
</tr>
<tr>
<td>ROW UPDTE</td>
<td>The number of rows updated.</td>
</tr>
</tbody>
</table>
Field Name: QW0018UP

ROW DELET
The number of rows deleted. If the delete was a mass delete, the indicator MASS is printed.

Field Name: QW0018DE

PAGES
The number of get page requests issued by the data manager to the buffer manager. Note that for an index scan the value includes the number of index pages scanned but not the number of index subpages scanned.

Field Name: QW0018PS

RI SCAN
The number of additional pages scanned for referential integrity.

Field Name: QW0018PR

RI DELET
The number of additional rows deleted for referential integrity.

Field Name: QW0018DR

ROW SKIP
The number of rows skipped due to an incompatible hold lock.

Field Name: QW0018SK
IFCID 019 - Edit Exit to Decode

This topic shows detailed information about “Record Trace - IFCID 019 - Edit Exit to Decode”.

Record trace - IFCID 019 - Edit Exit to Decode

The field labels shown in the following sample layout of “Record Trace - IFCID 019 - Edit Exit to Decode” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBID</td>
<td>The database ID. Deduced from the DB2 fields QW0019DB, QW0105DN or QW0107DN. When present, the database name is shown, otherwise the decimal identifier from QW0019DB is shown, or N/A when this value is 0. Field Name: RT0019DB</td>
</tr>
<tr>
<td>REC ID</td>
<td>The decimal identifier of the DB2 table OBID. Field Name: QW0019OB</td>
</tr>
<tr>
<td>OBID</td>
<td>The object ID. Deduced from the DB2 fields QW0019OB, QW0105TN or QW0107TN. When present, the name of the object is shown, otherwise the decimal identifier from QW0019OB is shown, or N/A when this value is 0. Field Name: RT0019OB</td>
</tr>
<tr>
<td>TIME</td>
<td>The time at which the exit was called. Field Name: QW0019TM</td>
</tr>
<tr>
<td>RETURN</td>
<td>The return code (EXPLRC1) from the user edit exit. Field Name: QW0019RT</td>
</tr>
<tr>
<td>REASON</td>
<td>The reason code (EXPLRC2) from the user edit exit. Field Name: QW0019RE</td>
</tr>
</tbody>
</table>
IFCID 020 - Lock Summary

This topic shows detailed information about “Record Trace - IFCID 020 - Lock Summary”.

This record has a variable format. It contains one data section for each table table
space section present in the record.

Record trace - IFCID 020 - Lock Summary

The field labels shown in the following sample layout of “Record Trace - IFCID 020
- Lock Summary” are described in the following section.

MAXNO: 12 SHARED: 0 EXCL: 0
DBID : FIJ1DB01 OBID : FIJS0010 MAX LOCK: 0
TABLESPACE TYPE: PARTIT.-SPL ESCALATED: 0
LOCK SIZE : TABLESPACE OR TABLE
HISTATE: INTENT SHARE PRESTATE: NO LOCK ESCALATION

MAXNO

The maximum number of page, row and LOB locks held concurrently for
the thread across all tables spaces and index spaces.

Field Name: QW0020TP

SHARED

The number of escalations to shared mode for the thread:
  • For segmented table spaces, the number of tables that have escalated
  • For partitioned table spaces using selective partition locking (SPL), the
    number of partitions that have escalated
  • For simple and partitioned table spaces, the number of table spaces that
    have escalated

Field Name: QW0020TS

EXCL

The number of escalations to exclusive mode for the thread:
  • For segmented table spaces, the number of tables that have escalated
  • For partitioned table spaces using selective partition locking (SPL), the
    number of partitions that have escalated
  • For simple and partitioned table spaces, the number of table spaces that
    have escalated

Field Name: QW0020TX

Table space sections

The record contains one data section for each relevant table space. These
sections are only printed if they are present in the record.

Field Name: QW0020N

DBID

The database ID. Deduced from the DB2 fields QW0020PD, QW0105DN or
QW0107DN.

When present, the database name is shown, otherwise the decimal
identifier from QW0020PD is shown, or N/A when this value is 0.

Field Name: RT0020DB
IFCID 020 - Lock Summary

OBID

The object ID. Deduced from the DB2 fields QW0020PP, QW0105TN or QW0107TN.

When present, the name of the object is shown, otherwise the decimal identifier from QW0020PP is shown, or N/A when this value is 0.

Field Name: RT0020OB

MAX LOCK

The maximum number of either page, row or LOB locks held by the thread.

Field Name: QW0020PL

ESCALATED

The number of escalations:

- For segmented table spaces, the number of tables that have escalated within the table space
- For partitioned table spaces using selective partition locking (SPL), the number of partitions that have escalated
- For table spaces using SPL, the number of partitions that have escalated

If the value in TABLESPACE TYPE is SIMPLE or PARTITIONED, this field is not printed.

Field Name: QW0020PC

LOCK SIZE

The lock size used.

Field Name: QW0020PR

HISTATE

The highest table space lock state. This field is printed for simple table spaces and partitioned table spaces not using SPL.

Field Name: QW0020PS

PRESTATE

The table space lock state before escalation. This field is printed for simple table spaces and partitioned table spaces not using SPL.

Field Name: QW0020PE
IFCID 021 - Lock Detail

This topic shows detailed information about “Record Trace - IFCID 021 - Lock Detail”.

Record trace - IFCID 021 - Lock Detail

The field labels shown in the following sample layout of “Record Trace - IFCID 021 - Lock Detail” are described in the following section.

**LOCK RES TYPE**

The locked resource type.

**Note:** For data sharing, SKELETON CURSOR TABLE LOCKING and SKELETON PACKAGE TABLE LOCK are LP-locks (an LP-lock has an L-lock component and a P-lock component).

**Field Name:** QW0021KT

**DBID**

The database ID. This field is not applicable if the value in LOCK RES TYPE is:

SKELETON CURSOR TABLE LOCKING
UTILITY SERIALIZATION LOCK
SKELETON PACKAGE TABLE LOCK
COLLECTION
BINDLOCK
ALTER BUFFER POOL
GROUP BUFFERPOOL START/STOP LOCK
GROUP BUFFER POOL LEV CASTOUT P-LOCK
CATMAINT MIGRATION LOCK
CATMAINT CONVERT CATALOG LOCK
CATMAINT CONVERT DIRECTORY LOCK

**Field Name:** QW0021KD

**OBID**

The object ID. This field is not applicable if the value in LOCK RES TYPE is:

SKELETON CURSOR TABLE LOCKING
UTILITY SERIALIZATION LOCK
SKELETON PACKAGE TABLE LOCK
COLLECTION
BINDLOCK
ALTER BUFFER POOL
GROUP BUFFERPOOL START/STOP LOCK
DDF CDB P-LOCK
GROUP BUFFER POOL LEV CASTOUT P-LOCK
DBD P-LOCK
CATMAINT MIGRATION LOCK
CATMAINT CONVERT CATALOG LOCK
CATMAINT CONVERT DIRECTORY LOCK

**Field Name:** QW0021KP
RESOURCE ID

The hexadecimal identifier of the small resource. If LOCK RES TYPE is:

DATA PAGE LOCKING
First 3 bytes are the page number

PARTITION LOCKING
Last byte is the partition number

INDEX PAGE LOCKING
First 3 bytes are the page number and the last byte is the subpage number

HASH ANCHOR LOCK
First 3 bytes are the page number and the last byte is the anchor point ID

CS-READ DRAIN
Last byte is the partition number (optional)

RR-READ DRAIN
Last byte is the partition number (optional)

WRITE DRAIN
Last byte is the partition number (optional)

ROW LOCK
First 3 bytes are the page number and the last byte is the row ID of the record

INDEX END OF FILE LOCK
Last byte is the partition number (optional)

PAGESET/PARTITION P-LOCK
First byte is the 1-based partition number (optional)

PAGE P-LOCK
First byte is the 1-based partition number (optional) and the last 3 bytes are the relative page number

PAGESET/PARTITION LEV CASTOUT P-LOCK
First byte is the 1-based partition number (optional)

Note: In large partitioned table spaces, the page number covers 4 bytes instead of 3.

For all other lock resource types, the resource ID is not applicable.

Field Name: QW0021KR

NAME

The plan name or collection name. This field is only printed if the value in LOCK RES TYPE is SKELETON CURSOR TABLE LOCKING or COLLECTION.

Deduced from the DB2 field QW0021KD, QW0021KP, and QW0021KR. when the locked resource type is skeleton cursor table locking.

Field Name: RT21NAME

COLL

The collection identifier. This field is only printed if the value in LOCK RES TYPE is SKELETON PACKAGE TABLE LOCK.
IFCID 021 - Lock Detail

The package identifier. This field is only printed if the value in LOCK RES TYPE is SKELETON PACKAGE TABLE LOCK.

The consistency token. This field is only printed if the value in LOCK RES TYPE is SKELETON PACKAGE TABLE LOCK.

Field Name: QW0021RN

BPID

The buffer pool ID. This field is only printed if the value in LOCK RES TYPE is:
ALTER BUFFER POOL
GROUP BUFFERPOOL START/STOP LOCK
PAGESET/PARTITION P-LOCK
PAGE P-LOCK
GROUP BUFFERPOOL LEV CASTOUT P-LOCK
PAGESET/PARTITION LEV CASTOUT P-LOCK

For ALTER BUFFER POOL, deduced from QW0021KD || QW0021KP.
For GROUP BUFFERPOOL START/STOP LOCK, deduced from QW0021KD || QW0021KP.
For PAGESET/PARTITION P-LOCK, deduced from QW0021P1.
For PAGE P-LOCK, deduced from QW0021P1.
For GROUP BUFFERPOOL LEV CASTOUT P-LOCK deduced from QW0021P1.
For PAGESET/PARTITION LEV CASTOUT P-LOCK, deduced from QW0021P1.

Field Name: RT21BPID

IRLM FUNC CODE

The IRLM function code.

Field Name: QW0021FC

RETURN TOKEN

The IRLM returned token.

Field Name: QW0021FT

REQUEST TOKEN

The lock request token. If the value in IRLM FUNC CODE is LOCK, this field shows "BLANK". If the value in IRLM FUNC CODE is UNLOCK or CHANGE, this field contains a 0 or a non-zero value. A 0 indicates that the lock name is used to identify the object that is to be unlocked or changed. A non-zero value is the same as the value in RETURN TOKEN. It associates the unlock or change request with the locked object.

Field Name: QW0021RT

LOCK STATE

The lock state.

Field Name: QW0021ST

DB2 TOKEN

The DB2 token which identifies the subsystem.

Field Name: QW0021TK
IRLM RETURN CODE

- **0**: The request completed successfully.
- **4**: The request completed successfully, but the lock state remained unchanged.
- **8**: The request completed unsuccessfully because of a system error or condition.
- **12**: The request completed unsuccessfully because of a logic error in the request.
- **16**: The request completed unsuccessfully because of an invalid request specification.
- **20**: The request completed unsuccessfully because IRLM resources are not available.

Field Name: QW0021RC

LOCK ATTRIBUTES

This field shows various lock attributes.

Field Name: QW0021FL

PROP TO XES

Indicates whether the request was propagated to XES by IRLM.

Field Name: QW0021Y1

ASYN TO XES

Indicates whether IRLM sent the request to XES asynchronously.

Field Name: QW0021Y2

LOCK DURATION

The lock duration:

- **MANUAL**: Varies depending on the ISOLATION parameter (QW0021DR=x'20')
- **MANUAL+1**: Temporary change of consistency level from CS to RR during bind and DDL (QW0021DR=x'21')
- **COMMIT**: Until commit (QW0021DR=x'40')
- **COMMIT+1**: Past commit; applies to locks needed to maintain the position for a cursor opened WITH HOLD (QW0021DR=x'41')
- **ALLOCATION**: Until deallocation (QW0021DR=x'60')
- **PLAN**: For the duration of the plan (QW0021DR=x'80')
- **UTIL**: For the duration of the utility execution (QW0021DR=x'81')
- **INTEREST**: Duration used for P-locks (QW0021DR=x'FE')
- **FREE ALL**: Until all locks are freed (QW0021DR=x'FF')
ICCID 021 - Lock Detail

N/A Not applicable for NOTIFY SUSPEND

Field Name: QW0021DR

REQUEST TYPE
Indicates whether it was a P-lock or L-lock request.

Field Name: QW0021Z1

IRLM RETURN SUBCODE
The IRLM return subcode.

Field Name: QW0021SC

PARENT TOKEN
The parent lock token for explicit hierarchical locking. This token is only significant when DB2 is a member of a data sharing group. If the value in this field is not 0, then this request is for a child of a parent that has already been locked. This value must match the RETURN TOKEN field of the previously locked parent. This field is only applicable if the value in IRLM FUNC CODE is LOCK.

Field Name: QW0021PT

GLOBAL/LOCAL
Indicates whether this is a global or local lock.

Field Name: QW0021GF

OWNER
The DB2 member name of either of the following:
- The owner of an incompatible retained lock on this resource that caused this request to be denied
- The owner of an incompatible held lock on this resource that caused this request to timeout

Field Name: QW0021SN

CACHED STATE
The cached state of the P-lock. This field is only applicable if the value in REQUEST TYPE is P-LOCK, and the value in LOCK RES TYPE is PAGESET/PARTITION P-LOCK.

Field Name: QW0021CS

LOCK HASH VALUE
The hash value of the locked resource.

Field Name: QW0021LH
IFCID 022 - Minibind

This topic shows detailed information about “Record Trace - IFCID 022 - Minibind”.

Minibind record shows information about mini plans, which are generated by the optimizer at bind and SQL prepare time. One mini plan is generated for each table and for each subselect block in the query. This means that if your query uses subqueries, more than one mini plan record is written.

Note:
- When interpreting this record, relate table and mini plans by table name.
- The order of the mini plans might not be the same as the order of the table as written in the SQL statement.
- When you are not sure about the accessing order of the tables, use EXPLAIN to get the query block number and plan number.
- This IFCID shows whether sequential prefetch is used.
- This mini plan block is written for each query and repeated for each subsequent subquery.
- If the query or subquery uses index scan (INDEX_NUMBER > 0), information is provided for each index used.

Record trace - IFCID 022 - Minibind

The field labels shown in the following sample layout of “Record Trace - IFCID 022 - Minibind” are described in the following section.

QUERYNO: 0
PLANNAME: ADB
COST: 615
PARALLELISM_DISABLED: N/A
QBLOCKNO: 1
COLLID: DSNDYNAMICSQLCACHE
PRODNAME: ADBMAIN
CONSISTENCY_TOKEN: X'262D553B00000050'
APPLNAME: 'BLANK'
WHEN_OPTIMIZE: README
OPT_HINT_IDENT: 'BLANK'
OPTIMIZE_HINTS_USED: NO
UNITS: 0
MILLI_SEC: 0
COST_CATEGORY: N/P
PARENT_Q_BLOCKNO: 0
MEMBER: SDA2
STATEMENT_TYPE: SELECT
TOOLSET: 2003/07/29 15:31:20.43
VERSION: N/P
REASON: TABLE_CARDINALITY / HAVING_CLAUSE

QUERYNO
The number identifying the statement to be prepared.

Field Name: QW0022QN

PLANNAME
The plan name or package ID.

Field Name: QW0022PN
COST
The relative cost of the SQL statement. It might not relate to the actual CPU or elapsed time for the query.
Field Name: QW0022OS

PARALLELISM_DISABLED
Indicates whether query parallelism is disabled by the resource limit facility (RLF) for dynamic queries:

NO The RLF does not affect this statement. (QW0022RP=x '00')
I/O ONLY Query I/O parallelism is disabled. (QW0022RP=x '01')
CP ONLY Query CP parallelism is disabled. (QW0022RP=x '02')
CP + I/O Query I/O and CP parallelism is disabled. (QW0022RP=x '03')
X Sysplex query parallelism is disabled. (QW0022RP=x '04')
X + I/O Sysplex query and query I/O parallelism is disabled. (QW0022RP=x '05')
X + CP Sysplex query and query CP parallelism is disabled. (QW0022RP=x '06')
YES The entire query parallelism (I/O, CP, and sysplex) is disabled. (QW0022RP=x '07')
N/A Query parallelism does not apply to this statement. (QW0022RP=x 'FF')

Field Name: QW0022RP

QBLOCKNO
The position of the query in the statement.
Field Name: QW0022QB

COLLID
The collection ID of the package.
Field Name: QW0022CI

PROGNAME
The name of the package containing the statement to be prepared.
Field Name: QW0022PG

CONSISTENCY_TOKEN
The consistency token.
Field Name: QW0022CT

APPLNAME
The name of the application plan.
Field Name: QW0022AL

WHEN_OPTIMIZE
Indicates when the access path of the SQL statement is optimized:

**DEFAULT**
- The access path is determined at bind time using default values.

**BIND**
- The access path is determined at bind time using default values, but it is reoptimized at runtime using values of input variables.

**RUN**
- The access path is determined at runtime using values of input variables.

**REOPT**
- The access path is reoptimized at runtime because the value of the host variable or parameter marker changes.

**Field Name:** QW0022RX

**OPT_HINT_IDENT**
- Access path hint value.

**Field Name:** QW0022QO

**OPTIMIZE_HINTS_USED**
- Indicates whether the query used access path hints.

**Field Name:** QW0022HT

**UNITS**
- Estimated processor cost in service units for the SQL statement.

**Field Name:** QW0022AS

**MILLI_SEC**
- Estimated processor cost in milliseconds for the SQL statement.

**Field Name:** QW0022CE

**COST_CATEGORY**
- The cost category for the statement can be one of the following:
  - **A** This SQL statement is a category A statement.
  - **B** This SQL statement is a category B statement.
  - **'BLANK'** Indicates that there is no processor cost estimate for this trace record.

**Field Name:** QW0022CC

**PARENT_Q_BLOCKNO**
- Parent query block number.

**Field Name:** QW0022PQ

**MEMBER**
- The member name of the DB2 that executed EXPLAIN. The column is blank if the DB2 subsystem was not in a data sharing environment when EXPLAIN was executed.

**Field Name:** QW0022GM

**STATEMENT_TYPE**
For each query block, the type of operation performed. For the outermost query, the statement type. Possible values:

SELECT
    SELECT
INSERT
    INSERT
UPDATE
    UPDATE
DELETE
    DELETE
SELUPD
    SELECT for UPDATE
DELCUR
    DELETE current of cursor
UPDCUR
    UPDATE current of cursor
CORSUB
    Correlated subquery
NCOSUB
    Noncorrelated subquery

Field Name: QW0022QT

TIMESTAMP
The timestamp at which the row is processed.
Field Name: QW0022TS

BIND TIME
The date and time at which the plan or package to which this statement belongs was bound.
Field Name: QW0022BT

VERSION
The version ID of the package.
Field Name: QW0022VN

PREDICATE #
If the REASON field has a value of REOPT, the predicate number that triggers the REOPT decision is shown.
Field Name: QW0022PD

REASON
Reason code for cost category B. This value is blank if the cost category is not B. Possible values are:

HOST VARIABLES
    If there are host variables, parameter markers, or special registers in range or between predicates.

TABLE CARDINALITY
    If the table cardinality is missing for one or more tables.
TRIGGERS
If there are insert, update, or delete triggers defined on the target table.

UDF
If there are user-defined functions referenced in the SQL statement.

REFERENTIAL CONSTRAINTS
If a table that is the target of a delete has referential constraints defined on it.

HAVING CLAUSE
If a having clause causes an SQL statement to be assigned to cost category B.

Field Name: QW0022RS

PLANNO
The plan number of the step in which the query is processed.

Field Name: QW0022PL

METHOD or NEXTSTEP
The join method used for the step.

Note: NEXTSTEP is shown if this field has one of the following values 0, 4, 8, 12, 1, 5, 9, 13, 2, 6, 10, 14, 3, 7, 11, or 15. Otherwise, METHOD is displayed.

Field Name: QW0022OD

SORTN_UNIQ
Indicates whether the new table is sorted to remove duplicate rows.

Field Name: QW0022UN

SORTC_UNIQ
Indicates whether the composite table is sorted to remove duplicate rows.

Field Name: QW0022N

DATABASE
The database ID.

Field Name: QW0022DD

SORTN_JOIN
Indicates whether the new table is sorted for a merge scan join or hybrid join. For a hybrid join, this is a sort of the RID list.

Field Name: QW0022IN

SORTC_JOIN
Indicates whether the composite table is sorted for a nested loop join, merge scan join, or hybrid join.

Field Name: QW0022J

OBJECT
The internal ID of the table in hexadecimal (2 bytes). Use this value to match column “OBID” in SYSIBM.SYSTABLES to find the name of the table. For example, X’2A’ is 42, which is table SYSDATABASE.
### IFCID 022 - Minibind

Field Name: QW0022OB

**ACCESSSTYPE**

The method of accessing the new table. N/P is printed if there is no access type.

Field Name: QW0022YP

**SORTN_ORDERBY**

Indicates whether the new table is sorted for ORDER BY.

Field Name: QW0022DB

**SORTC_ORDERBY**

Indicates whether the composite table is sorted for ORDER BY.

Field Name: QW00222O

**CREATOR**

The creator of the new table accessed in this step.

Field Name: QW0022CR

**PAGE_RANGE**

Whether the table qualifies for page range screening, so that plans scan only the partitions that are needed. Y = Yes; N = No.

Field Name: QW0022PR

**SORTN_GROUPBY**

Indicates whether the new table is sorted for GROUP BY.

Field Name: QW0022PB

**SORTC_GROUPBY**

Indicates whether the composite table is sorted for GROUP BY.

Field Name: QW00222G

**TNAME**

The name of the table accessed in this step, without qualifier. This field is blank if a view is used instead of a real table.

Field Name: QW0022TN

**JOIN_TYPE**

The type of join:

- **F** FULL OUTER JOIN
- **L** LEFT OUTER JOIN
- **S** STAR JOIN
- **blank** INNER JOIN or no join

RIGHT OUTER JOIN converts to a LEFT OUTER JOIN when you use it, so that JOIN_TYPE contains L.

Field Name: QW0022JT

**SORTN_PGROUP_ID**

The parallel group identifier for the parallel sort of the new table.
Field Name: QW0022P6
SORTC_PGROUP_ID
The parallel group identifier for the parallel sort of the composite table.
Field Name: QW0022P7
CORRELATION_NAME
The correlation name of a table or view that is specified in the statement. If there is no correlation name, then the column is blank.
Field Name: QW0022CN
MERGE_JOIN_COLS
The number of columns that are joined during a merge scan join (Method=2).
Field Name: QW0022JC
ACCESS_DEGREE
The number of parallel tasks or operations activated by a query.
Field Name: QW0022P1
JOIN_DEGREE
The number of parallel tasks or operations used in joining the composite table with the new table.
Field Name: QW0022P3
TSLOCKMODE
Indicates the lock mode to be acquired on the new table or its table space.
If the isolation can be determined at bind time, possible values are:
IS    Intent share lock
IX    Intent exclusive lock
S     Share lock
U     Update lock
X     Exclusive lock
SIX   Share with intent exclusive lock
N     UR isolation, no lock
If the isolation cannot be determined at bind time, the lock mode determined by the isolation at run time is shown by the following values:
NS    For UR isolation: no lock. For CS or RR isolation: an S lock.
NIS   For UR isolation: no lock. For CS or RR isolation: an IS lock.
NSS   For UR isolation: no lock. For CS isolation: an IS lock. For RR isolation: an S lock.
SS    For UR or CS isolation: no lock. For RR isolation: an S lock.
The data in this column is right-justified.
Field Name: QW0022LM
PARALLELISM_MODE
IFCID 022 - Minibind

The kind of parallelism, if any, that is used at bind time:
I   Query I/O parallelism
C   Query CP parallelism
X   Sysplex query parallelism

Field Name: QW0022PM

ACCESS_PGROUP_ID
The ID of the parallel group for accessing the new table.
Field Name: QW0022P2

JOIN_PGROUP_ID
The ID of the parallel group for joining the composite table with the new table.
Field Name: QW0022P4

AGGREGATE_FUNCT
Indicates when an SQL column function is evaluated. Possible values are:
R   Column function is evaluated during data retrieval.
S   Column function is evaluated during SORT.
Field Name: QW0022Z

INDEX_NUMBER
Number of index access operations.
Field Name: QW0022MN

PREFETCH
The number of PREFETCH requests.
Field Name: QW0022EF

DIRECT_ROW_ACC
Indicates whether DB2 can use direct row access to a table row without a table space or index scan:
YES  Direct row access was used
NO   Direct row access was not used
Field Name: QW0022PA

PAGES_FOR_TABLE
The number of pages for the table. A value of "-1" indicates that statistics are not available.
Field Name: QW0022NP

TAB/Cardinality
Table cardinality in floating point.
Field Name: QW0022CY

STARJOIN
Indicates whether star join was used, possible values are:
YES  Star join was used
NO   Star join was not used
Field Name: QW0022SJ

TABLE_TYPE
The table type can be:
T    Table
F    Table function
W    Workfile
Q    Table queue (not materialized)
Field Name: QW0022TT

INDEXONLY
Indicates what kind of prefetch of the data is used:
SEQ  Sequential prefetch
LIST List prefetch
NO   No prefetch
Field Name: QW0022XO

MATCHCOLS
The number of index keys used in an index scan. This field is 0 if either no index is used or an index is used that has no matching columns.
Field Name: QW0022XM

MIXOPSEQ
The sequence number of a step in a multiple index operation.
Field Name: QW0022MS

PREFETCH_INDEX
Indicates whether data pages are to be read in advance by a prefetch.
Field Name: QW0022XF

OPERATION
The type of index access operation.
Field Name: QW0022MO

ACCESS_NAME
The index name. This field applies only to index scans. N/A is printed for table space scans or when no index is used.
Field Name: QW0022XN

ACCESS_CREATOR
The index creator.
Field Name: QW0022XC
IFCID 023 - Utility Start

This topic shows detailed information about “Record Trace - IFCID 023 - Utility Start”.

Record trace - IFCID 023 - Utility Start

The field labels shown in the following sample layout of “Record Trace - IFCID 023 - Utility Start” are described in the following section.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBID 1</td>
<td>DBID</td>
</tr>
<tr>
<td>UTILITY NAME</td>
<td>The utility name. Field Name: QW0023NM</td>
</tr>
<tr>
<td>OBID 2</td>
<td>OBID</td>
</tr>
<tr>
<td>UTILITY PHASE</td>
<td>The phase name of the utility. Field Name: QW0023PH</td>
</tr>
<tr>
<td>RQSTASK</td>
<td>The number of requested subtasks. Field Name: QW0023R1</td>
</tr>
<tr>
<td>UTILID 3</td>
<td>UTILITY ID</td>
</tr>
<tr>
<td>DBNAME</td>
<td>The database name.</td>
</tr>
</tbody>
</table>
IFCID 023 - Utility Start

Field Name: QW0023DB

OBJECT NAME
The object name.

Field Name: QW0023PD

REORG KEEPDICTIONARY
The reorg utility is specified with the KEEPDICTIONARY keyword.

Field Name: QW0023D1

LOAD KEEPDICTIONARY
The load utility is specified with the KEEPDICTIONARY keyword.

Field Name: QW0023B1

COPY CONCURRENT
The copy utility is specified with the CONCURRENT keyword.

Field Name: QW0023A1

REBUILD REUSE
The rebuild utility is specified with the REUSE keyword.

Field Name: QW0023F1

REORG REUSE
The reorg utility is specified with the REUSE keyword.

Field Name: QW0023D2

LOAD REUSE
The load utility is specified with the REUSE keyword.

Field Name: QW0023B2

COPY SHRLEVEL CHANGE
The copy utility is specified with the SHRLEVEL(CHANGE) keyword.

Field Name: QW0023A2

REBUILD SORTKEYS
The rebuild utility is specified with the SORTKEYS keyword.

Field Name: QW0023F2

REORG LOG NO
The reorg utility is specified with the LOG(NO) keyword.

Field Name: QW0023D3

LOAD LOG NO
The load utility is specified with the LOG(NO) keyword.

Field Name: QW0023B3

COPY PARALLEL
The copy utility is specified with the PARALLEL keyword.

Field Name: QW0023A3
REBUILD STATISTICS
The rebuild utility is specified with the STATISTICS keyword.
Field Name: QW0023F3

REORG SORTKEYS
The reorg utility is specified with the SORTKEYS keyword.
Field Name: QW0023D4

LOAD SORTKEYS
The load utility is specified with the SORTKEYS keyword.
Field Name: QW0023B4

COPY CHECKPAGE
The copy utility is specified with the CHECKPAGE keyword.
Field Name: QW0023A4

REBUILD WORKDDN
The rebuild utility is specified with the WORKDDN keyword.
Field Name: QW0023F4

REORG SORTDATA
The reorg utility is specified with the SORTDATA keyword.
Field Name: QW0023D5

LOAD SHRLEVEL CHANGE
The load utility is specified with the SHRLEVEL(CHANGE) keyword.
Field Name: QW0023B5

REORG NOSYSREC
The reorg utility is specified with the NOSYSREC keyword.
Field Name: QW0023D6

LOAD COPYDDN
The load utility is specified with the COPYDDN keyword.
Field Name: QW0023B6

RECOVER REUSE
The recover utility is specified with the REUSE keyword.
Field Name: QW0023A7

RUNSTATS SAMPLE
The runstats utility is specified with the SAMPLE keyword.
Field Name: QW0023G1

REORG SHRLEVEL CHANGE
The reorg utility is specified with the SHRLEVEL(CHANGE) keyword.
Field Name: QW0023D7

LOAD STATISTICS
IFCID 023 - Utility Start

The load utility is specified with the STATISTICS keyword.

**Field Name:** QW0023B7

**RECOVER PARALLEL**

The recover utility is specified with the PARALLEL keyword.

**Field Name:** QW0023A8

**RUNSTATS SHRLEVEL CHANGE**

The runstats utility is specified with the SHRLEVEL(CHANGE) keyword.

**Field Name:** QW0023G2

**REORG SHRLEVEL REFERENCE**

The reorg utility is specified with the SHRLEVEL(REFERENCE) keyword.

**Field Name:** QW0023D8

**LOAD PART INDDN**

The load utility is specified with the PART(INDDN) keyword.

**Field Name:** QW0023B8

**REORG COPYDDN**

The reorg utility is specified with the COPYDDN keyword.

**Field Name:** QW0023E1

**UNLOAD SHRLEVEL REFERENCE**

The unload utility is specified with the SHRLEVEL(REFERENCE) keyword.

**Field Name:** QW0023H1

**REORG STATISTICS**

The reorg utility is specified with the STATISTICS keyword.

**Field Name:** QW0023E2

**UNLOAD SHRLEVEL CHANGE ISOLATION CS**

The unload utility is specified with the SHRLEVEL(CHANGE ISOLATION CS) keyword.

**Field Name:** QW0023H2

**REORG FASTSWITCH**

The reorg utility is specified with the FASTSWITCH keyword.

**Field Name:** QW0023E3

**UNLOAD SHRLEVEL CHANGE ISOLATION UR**

The unload utility is specified with the SHRLEVEL(CHANGE ISOLATION UR) keyword.

**Field Name:** QW0023H3
IFCID 024 - Utility Change

This topic shows detailed information about “Record Trace - IFCID 024 - Utility Change”.

Record trace - IFCID 024 - Utility Change

The field labels shown in the following sample layout of “Record Trace - IFCID 024 - Utility Change” are described in the following section.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBID</td>
<td>The database ID. Deduced from the DB2 fields QW0024DB, QW0105DN or QW0107DN. When present, the database name is shown, otherwise the decimal identifier from QW0024DB is shown, or N/A when this value is 0. Field Name: RT0024DB</td>
</tr>
<tr>
<td>UTILITY NAME</td>
<td>The utility name. Field Name: QW0024NM</td>
</tr>
<tr>
<td>OBID</td>
<td>The object ID. Deduced from the DB2 fields QW0023PD, QW0105TN or QW0107TN. When present, the name of the object is shown, otherwise the decimal identifier from QW0024OB is shown, or N/A when this value is 0. Field Name: RT0024OB</td>
</tr>
<tr>
<td>UTILITY PHASE</td>
<td>The phase name of the utility. Field Name: QW0024PH</td>
</tr>
<tr>
<td>ITEMS</td>
<td>The number of items processed by the utility. Field Name: QW0024DN</td>
</tr>
<tr>
<td>UTILITY ID</td>
<td>The identifier of the utility. Field Name: QW0024ID</td>
</tr>
<tr>
<td>DBNAME</td>
<td>The database name. Field Name: QW0024NA</td>
</tr>
<tr>
<td>OBJECT NAME</td>
<td>The table space name or index name. Field Name: QW0024PN</td>
</tr>
</tbody>
</table>
PART/DATASET#

The number of the partition or data set if the utility is operating on one partition or data set. Otherwise, the value in this field is 0.

Field Name: QW0024PT
This topic shows detailed information about “Record Trace - IFCID 025 - Utility End”.

Record trace - IFCID 025 - Utility End

The field labels shown in the following sample layout of “Record Trace - IFCID 025 - Utility End” are described in the following section.

DBID

The database ID. Deduced from the DB2 fields QW0025DB, QW0105DN or QW0107DN.

When present, the database name is shown, otherwise the decimal identifier from QW0025DB is shown, or N/A when this value is 0.

Field Name: RT0025DB

UTILITY NAME

The utility name.

Field Name: QW0025NM

OBID

The object ID. Deduced from the DB2 fields QW0025PD, QW0105TN or QW0107TN.

When present, the name of the object is shown, otherwise the decimal identifier from QW0025OB is shown, or N/A when this value is 0.

Field Name: RT0025OB

UTILITY PHASE

The phase name of the utility.

Field Name: QW0025PH

ITEMS

The phase name of the utility.

Field Name: QW0025DN

UTILITY ID

The identifier of the utility.

Field Name: QW0025ID

JOBNAME

The job name of the utility.
Field Name: QW0025JN

STEPNAME

The step name of the utility job.

Field Name: QW0025JS

ELAPSED TIME

The utility elapsed time at termination. This field and the following time fields are in time-of-day format. If this field contains binary zeroes, no data is available for this field or for the following time fields. For example, this is the case for subphase termination records.

Field Name: QW0025UE

SUBTASKS

The final subtask count.

Field Name: QW0025R1

DFSORT

DFSORT was invoked at least once. Possible values are Y or N.

Field Name: QW0025DF

DB2SORT

DB2 SORT was invoked at least once. Possible values are Y or N.

Field Name: QW0025DS

DATA Sorts

The number of parallel data sorts.

Field Name: QW0025DA

INDEX Sorts

The number of parallel index sorts.

Field Name: QW0025IX

OTHER SORTS

The number of other sorts.

Field Name: QW0025OS

SHRLEVEL

The SHRLEVEL value of the utility. Possible values are: NONE, REFERENCE, CHANGE, or N/A.

Field Name: QW0025SL

CPU TIME

The CPU time of the utility.

Field Name: QW0025UC

ZIIP TIME

The total utility ZIIP time (if Accounting class 1 trace is activated).

Field Name: QW0025UZ

SORT CPU
The Sort CPU time.

Field Name: QW0025SC

SORT ZIIP

The Sort ZIIP time (if provided by the Sort program).

Field Name: QW0025SZ
This topic shows detailed information about “Record Trace - IFCID 026 - IBM Service Record”.

This record is for IBM service use.
IFCID 027 - Sort Workfile Records

This topic shows detailed information about “Record Trace - IFCID 027 - Sort Workfile Records”.

Record trace - IFCID 027 - Sort Workfile Records

The field labels shown in the following sample layout of “Record Trace - IFCID 027 - Sort Workfile Records” are described in the following section.

<table>
<thead>
<tr>
<th>RECORDS IN NEW WORKFILE ......: 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPARSE INDEX SPACE USED (KB) : 1</td>
</tr>
<tr>
<td>ESTIMATED SIZE ALL IN-MEM (KB): 39</td>
</tr>
<tr>
<td>KEY SIZE SP IDX (BYTES) : 88</td>
</tr>
<tr>
<td>CURRENT IDX IN PROCESS : 1</td>
</tr>
<tr>
<td>TYPE OF RECORD ............: H</td>
</tr>
</tbody>
</table>

RECORDS IN NEW WORKFILE

The number of records in the new work file.

Field Name: QW0027NR

SPARSE INDEX SPACE USED (KB)

The size of the in-memory work file in kilobytes (KB).

Field Name: QW0027OZ

ESTIMATED SIZE ALL IN-MEM (KB)

The APS estimated size of all sparse indexes in a query if they are all in the in-memory part (in KB).

Field Name: QW0027TZ

ESTIMATED RECORDS IN SP INDEX

The APS estimated number of records in the current sparse index.

Field Name: QW0027IR

SKIP FACTOR

The skip factor if sparse index records are found in the work file. It shows a value of 1 if found in the in-memory part.

Field Name: QW0027SF

RECORDS IN IN-MEMORY PART

The number of records in the in-memory part of the sparse index.

Field Name: QW0027IE

RECORDS IN WORKFILE PART

The number of records in the work-file part of the sparse index.

Field Name: QW0027WE

DATA AREA SIZE (BYTES)

The data area size for a sparse index (in bytes).

Field Name: QW0027DS

KEY SIZE SP IDX (BYTES)
IFCID 027 - Sort Workfile Records

- The key size for a sparse index (in bytes).
  
  Field Name: QW0027KS

TOTAL NR OF IDX IN QUERY
- The total number of sparse indexes in the query.
  
  Field Name: QW0027TS

CURRENT IDX IN PROCESS
- The current sparse index that is processed.
  
  Field Name: QW0027SC

TYPE OF RECORD
- The type of record. Possible values are:

  B  Indicates that a sparse index combination of hash and work file is used (both in-memory and physical work file).
  
  H  Indicates a sparse index hash is used (in-memory work file only).
  
  O  Indicates that a sparse index binary is used (in-memory work file only).
  
  S  Indicates that no sparse index is used because of storage constraints.
  
  T  Indicates that a sparse index work-file is used.
  
  W  Indicates that no sparse index is used.

  Field Name: QW0027SP
IFCID 028 - Sort Phase Detail

This topic shows detailed information about “Record Trace - IFCID 028 - Sort Phase Detail”.

Record trace - IFCID 028 - Sort Phase Detail

The field labels shown in the following sample layout of “Record Trace - IFCID 028 - Sort Phase Detail” are described in the following section.

OMEGUSER 'BLANK' RRS 15:15:38.81346278 991294 1 28 SORT PHASE DETAIL
KOEPLAN 'BLANK' N/P 15:15:38.81346278 991294 1 28 SORT PHASE DETAIL

WORKFILES REQ

The number of work files requested from the buffer manager at the beginning of each merge pass (MVS/ESA 3.1.3). It is valid if TYPE equals S.

If this field is greater than WORKFILES ACQ, there is another merge pass. If both fields are equal, this is the last or only merge pass.

Field Name: QW0028WA

TYPE

The type of IFCID 28. It indicates the phase when the IFCID 28 record is issued. Valid values are:

I The end of the input phase
S The start of a merge pass
E The end of a merge pass
Z The start of output work file partitioning
W During the output work file partitioning
X The end of output work file partitioning
K The start of last merge pass partitioning
M During last merge pass partitioning
L The end of last merge pass partitioning
T The start of one record partitioning
O During one record partitioning
U The end of one record partitioning

Field Name: QW0028NP
IFCID 028 - Sort Phase Detail

V The start of presorted records partitioning
P During presorted records partitioning
Y The end of presorted records partitioning

Field Name: QW0028TY

PASS
The current merge pass. It is issued at the end of the merge pass and, therefore, valid if TYPE equals E.
Field Name: QW0028MP

WORKFILES ACQ
The number of work files actually acquired from the buffer manager at the beginning of each merge pass (MVS/ESA 3.1.3). It is valid if TYPE equals S.
Field Name: QW0028WG

PARALLELISM DEGREE
The partition work file number. The value in this field is 0 if partitioning is not requested. If partitioning is requested, the value can be from 1 to n, where n is the degree of parallelism. It is valid if TYPE equals Z, W, X, K, M, L, T, O, U, V, P, or Y.
Field Name: QW0028PW

WORKFILE RECORDS
The number of records in the partition work file. It is valid if TYPE equals Z, W, X, K, M, L, T, O, U, V, P, or Y.
Field Name: QW0028PN

RECS SORTED AFT INS PHASE
The number of records sorted into work files after the sort input phase.
Field Name: QW0028NR

MULTIPLE DISTINCT SORTS
Total number of multiple distinct sorts.
Field Name: QW0028DS

MULTIPLE DISTINCT READ
The number of records read into a group at the start of the GROUPBY phase for a multiple distinct sort.
Field Name: QW0028DR

MULTIPLE DISTINCT GROUPS
The total number of multiple distinct sort groups.
Field Name: QW0028DG

CURRENT MULTIPLE DISTINCT SORT
The multiple distinct sort currently being processed.
Field Name: QW0028DC
IFCID 029 - EDM Request Start

**Record trace - IFCID 029 - EDM Request Start**

The field labels shown in the following sample layout of “Record Trace - IFCID 029 - EDM Request Start” are described in the following section.

**EDM REQUEST---> NETWORKID: DEIBMIPS LUNAME: IPSAU851 LUWSEQ: 1
START EDMID CT PLAN HSRTEP2L
RDS X'80000000' SEQNO 1 CT LGTH 120
EDM REQUEST---> 'BLANK'
START NETWORKID: DEIBMIPS LUNAME: IPSAU851 LUWSEQ: 1
EDMID CT PLAN HSRTEP2L
RDS X'00000001' SEQNO 0 CT LGTH 3560
EDM REQUEST---> 'BLANK'
START NETWORKID: DEIBMIPS LUNAME: IPSAU851 LUWSEQ: 1
EDMID PT
LOCATION N/P
COLLECTION HSRTEP2VL1XXXXXXXXX2XXXXXXXXX3XXXXXXXXX4X
XXXXXXXX5XXXXXXXXX6XXXXXXXXX7XXXXXXXXX8XX
XXXXXXXX9XXXXXXXXX0XXXXXXXXX1XXXXXXXXX2XXX
XXXXZ
PACKAGE ID DSN@EP2L@@@@@@@@@@
CONSISTENCY TOKEN X'0000000000000000'
RDS X'00000000' SEQNO 0 PT LGTH 120
QW0029SV X'0000'
EDM REQUEST---> NETWORKID: DEIBMIPS LUNAME: IPSAU851 LUWSEQ: 1
EDMID CT PLAN HSRTEP2L
RDS X'00000000' SEQNO 1 CT LGTH 120
EDM REQUEST---> 'BLANK'

**EDMID**

The type of request:
- **DB** Database descriptor
- **CT** Cursor table
- **PT** Package table

**Field Name:** QW0029ID

**DBID**

The database ID. Deduced from the DB2 fields QW0029DB, QW0105DN or QW0107DN.

When present, the database name is shown, otherwise the decimal identifier from QW0029DB is shown, or N/A when this value is 0.

**Field Name:** RT0029DB

**DB LGTH**

The length of the section associated with this DBD.

**Field Name:** QW0029DL

**PLAN**

The plan name for the CT or XT request.

**Field Name:** QW0029PL

**RDS**

The RDS identifier number. Special cases are:
IFCID 029 - EDM Request Start

x'00000001'
   SKCT header
x'FFFFFFFE'
   SKCT directory

Field Name: QW0029KN

SEQNO
   The sequence number within the RDS number. This is QW0029SN for CT or QW0029GN for PT.
   Field Name: RT0029SN

CT LGTH
   The length of the CT or XT sections in bytes.
   Field Name: QW0029CL

LOCATION
   The name of the package location. This field shows 'BLANK' if the local location name is not defined.
   Field Name: QW0029LN

COLLECTN
   The collection identifier of the package.
   Field Name: QW0029CI

PCKG ID
   The package identifier.
   Field Name: QW0029PI

CONSISTENCY TOKEN
   The consistency token of the package.
   Field Name: QW0029CT

PT LGTH
   The length of the PT section in bytes.
   Field Name: QW0029GL
Record trace - IFCID 030 - EDM Request End

The field labels shown in the following sample layout of “Record Trace - IFCID 030 - EDM Request End” are described in the following section.

EDM REQUEST< -- NETWORKID: DEIBMIPS LUNAME: IPSAU851 LUWSEQ: 1
END EDMID CT PLAN HSRTEP2L
RDS X'FFFFFFFE' SEQNO 2 CT CALLS 1
EDM REQUEST< -- 'BLANK'
END NETWORKID: DEIBMIPS LUNAME: IPSAU851 LUWSEQ: 1
EDMID CT PLAN HSRTEP2L
RDS X'00000001' SEQNO 2 CT CALLS 1
EDM REQUEST< -- 'BLANK'
END NETWORKID: DEIBMIPS LUNAME: IPSAU851 LUWSEQ: 1
EDMID PT
LOCATION N/P
COLLECTION HSRTEP2VL1XXXXXXXXX2XXXXXXXXX3XXXXXXXXX4X
XXXXXXXXX5XXXXXXXXX6XXXXXXXXX7XXXXXXXXX8XX
XXXXXXXXX9XXXXXXXXX0XXXXXXXXX1XXXXXXXXX2XXX
XXXXXZ
PACKAGE ID DSN0EP2L@0800000
CONSISTENCY TOKEN X'0000000000000000'
RDS X'00000000' SEQNO 0 PT CALLS 1
QW0030SV X'0000'

EDMID

The type of request:

DB Database descriptor
CT Cursor table
PT Package table

Field Name: QW0030ID

DBID

The database ID. Deduced from the DB2 fields QW0030DB, QW0105DN, or QW0107DN.

When present, the database name is shown, otherwise the decimal identifier from QW0030DB is shown, or N/A when this value is 0.

Field Name: RT0030DB

DB CALLS

The number of calls to the data manager for the DBD.

Field Name: QW0030DC

PLAN

The plan name for the CT or XT request.

Field Name: QW0030PL

RDS

The RDS identifier number for CT. Special cases are:
RDS
The RDS identifier number for PT. Special cases are:

SEQNO
The sequence number within the RDS number. This is QW0030SN for CT or QW0030GN for PT.

LOCATION
The name of the package location. This field shows "BLANK" if the local location name is not defined.

COLLCTN
The collection identifier of the package.

PCKG ID
The package identifier.

CONSISTENCY TOKEN
The consistency token of the package.

PT CALLS
The number of calls to the data manager for PT.
This topic shows detailed information about “Record Trace - IFCID 031 - EDM Full”.

**Record trace - IFCID 031 - EDM Full**

The field labels shown in the following sample layout of “Record Trace - IFCID 031 - EDM Full” are described in the following section.

```
EDM FULL --> NETWORKID: DEIBMIPS LUNAME: IPSAU851 LUWSEQ: 1
EDMID CT PLAN PLANNAM1
RDS X'00001267' SEQNO 1 CT LGTH 16
'BLANK'
EDM FULL --> NETWORKID: DEIBMIPS LUNAME: IPSAU851 LUWSEQ: 1
EDMID PT
LOCATION LOCATION01XXXXXXXXX2XXXXXXXXX3XXXXXXXXX4
XXXXXXXX5XXXXXXXXX6XXXXXXXXX7XXXXXXXXX8XX
XXXXXXXX9XXXXXXXXX0XXXXXXXXX1XXXXXXXXX2XX
XXXXXXXXZ
COLLECTION COLLECTION01XXXXXXXXX2XXXXXXXXX3XXXXXXXXX4
XXXXXXXX5XXXXXXXXX6XXXXXXXXX7XXXXXXXXX8XX
XXXXXXXX9XXXXXXXXX0XXXXXXXXX1XXXXXXXXX2XX
XXXXXXXXZ
PACKAGE ID PACKAGE001XXXXXXXXX2XXXXXXXXX3XXXXXXXXX4
XXXXXXXX5XXXXXXXXX6XXXXXXXXX7XXXXXXXXX8XX
XXXXXXXX9XXXXXXXXX0XXXXXXXXX1XXXXXXXXX2XX
XXXXXXXXZ
CONSISTENCY TOKEN X'C3D6D5E2E3D6D2F1'
RDS X'FFFFFFFE' SEQNO 66 PT LGTH 120
QW0031SV X'E7E7'
EDM FULL --> 'BLANK'
NETWORKID: DEIBMIPS LUNAME: IPSAU851 LUWSEQ: 1
EDMID PT
LOCATION LOCATION01XXXXXXXXX2XXXXXXXXX3XXXXXXXXX4
XXXXXXXX5XXXXXXXXX6XXXXXXXXX7XXXXXXXXX8XX
XXXXXXXX9XXXXXXXXX0XXXXXXXXX1XXXXXXXXX2XX
XXXXXXXXZ
COLLECTION COLLECTION01XXXXXXXXX2XXXXXXXXX3XXXXXXXXX4
XXXXXXXX5XXXXXXXXX6XXXXXXXXX7XXXXXXXXX8XX
XXXXXXXX9XXXXXXXXX0XXXXXXXXX1XXXXXXXXX2XX
XXXXXXXXZ
PACKAGE ID PACKAGE001XXXXXXXXX2XXXXXXXXX3XXXXXXXXX4
XXXXXXXX5XXXXXXXXX6XXXXXXXXX7XXXXXXXXX8XX
XXXXXXXX9XXXXXXXXX0XXXXXXXXX1XXXXXXXXX2XX
XXXXXXXXZ
CONSISTENCY TOKEN X'C3D6D5E2E3D6D2F1'
RDS X'FFFFFFFE' SEQNO 66 PT LGTH 120
QW0031SV X'E7E7'

EDMID

The type of request:
- **DB** Database
- **CT** Cursor
- **XT** DBD extension
- **PT** Package table

**Field Name:** QW0031ID

**DBID**

The database ID. Deduced from the DB2 fields QW0031DB, QW0105DN or QW0107DN.
When present, the database name is shown, otherwise the decimal identifier from QW0031DB is shown, or N/A when this value is 0.

Field Name: RT0031DB

DB LGTH
The length of the section associated with this DBD.
Field Name: QW0031DL

PLAN
The plan name for the CT or XT request.
Field Name: QW0031PL

RDS
The RDS identifier number for CT. Special cases are:
x'00000001'
   SKCT header
x'FFFFFFFFE'
   SKCT directory
Field Name: QW0031RN

RDS
The RDS identifier number for PT. Special cases are:
x'00000001'
   SKCT header
x'FFFFFFFFE'
   SKCT directory
Field Name: QW0031KN

SEQNO
The sequence number within the RDS number. This is QW0031SN for CT or QW0031GN for PT.
Field Name: R0031SN

CT LGTH
The length of the CT or XT sections in bytes.
Field Name: QW0031CL

LOCATION
The name of the package location. This field shows “BLANK” if the local location name is not defined.
Field Name: QW0031LN

COLLCTN
The collection identifier of the package.
Field Name: QW0031CI

PCKG ID
The package identifier.
Field Name: QW0031PI
IFCID 031 - EDM Full

CONSISTENCY TOKEN
The consistency token of the package.

Field Name: QW0031CT

PT LGTH
The length of the PT section in bytes.

Field Name: QW0031GL
IFCID 032 - Log Wait Start

This topic shows detailed information about “Record Trace - IFCID 032 - Log Wait Start”.

Record trace - IFCID 032 - Log Wait Start

The field labels shown in the following sample layout of “Record Trace - IFCID 032 - Log Wait Start” are described in the following section.

FUNC TYPE: WFRC

QW0032RB 155344864

FUNC TYPE

The function type or request type:

WFRC  Write force request
ARC  Archive log command
''  Normal write-force request

Deduced from the DB2 fields QW0032FT and QW0032RT.

Field Name: QW0032FT
IFCID 033 - IBM Service Record

This topic shows detailed information about “Record Trace - IFCID 033 - IBM Service Record”.

This record is for IBM service use.
IFCID 034 - Log Read Start

This topic shows detailed information about “Record Trace - IFCID 034 - Log Read Start”.

Record trace - IFCID 034 - Log Read Start

The field labels shown in the following sample layout of “Record Trace - IFCID 034 - Log Read Start” are described in the following section.

DSID : XXXXXXXX ACE : ZZ9
WAIT TIME TYPE: ACTIVE LOG READ
QW0034HR: X'HHHH' QW0034LR: X'HHHHHHHH'

DSID

The data set identifier of the log manager.

Field Name: QW0034DI

WAIT TIME TYPE

The type of wait time. Possible values are:

• ACTIVE LOG READ
• ACTIVE LOG PREFETCH READ
• BSDS READ
• PEER-BSDS READ

Otherwise, a hexadecimal value is shown.

Field Name: QW0034TY

ACE

The relative number of the agent control element address in the ACE cross-reference table. This table is printed at the end of each location for every trace specified.

Field Name: QW0034AC
Record trace - IFCID 035 - Log Read End

The field labels shown in the following sample layout of “Record Trace - IFCID 035 - Log Read End” are described in the following section.

**RET**

The return code.

*Field Name:* QW0035RT

**ACE**

The relative number of the agent control element address in the ACE cross-reference table. This table is printed at the end of each location for every trace specified.

*Field Name:* QW0035AC
IFCID 036 - Log Non I/O Start

This topic shows detailed information about “Record Trace - IFCID 036 - Log Non I/O Start”.

Record trace - IFCID 036 - Log Non I/O Start

The field labels shown in the following sample layout of “Record Trace - IFCID 036 - Log Non I/O Start” are described in the following section.

**DSID**

The data set identifier of the log manager.

**Field Name:** QW0036DI

**EVENT ID**

The event identifier:

- **ALLC** Allocation
- **DTAU** Data set unavailable
- **OPEN** Open
- **CLOS** Close
- **DEAL** Deallocate
- **CLOC** Wait for the catalog to be located
- **WTOR** Wait for reply from write-to-operator
- **HSMR** Wait for HSM recall
- **UUNI** Wait for unavailable tape unit
- **URST** Wait for unavailable reader service task
- **MDSV** Wait for multi-data set volume
- **POSI** Wait for tape volume positioning

**Field Name:** QW0036EI

**REQUEST TYPE**

The request type:

- **ALLD** Demand allocation
- **ALLL** Look ahead (premount) allocation

**Field Name:** QW0036RT

**ACE**

The relative number of the agent control element address in the ACE cross-reference table. This table is printed at the end of each location for every trace specified.

**Field Name:** QW0036AC
IFCID 037 - Log Non I/O End

This topic shows detailed information about “Record Trace - IFCID 037 - Log Non I/O End”.

Record trace - IFCID 037 - Log Non I/O End

The field labels shown in the following sample layout of “Record Trace - IFCID 037 - Log Non I/O End” are described in the following section.

```
RET          0  ACE          1
QW0037RC     0
RET

The return code.

Field Name: QW0037RT

ACE

The relative number of the agent control element address in the ACE cross-reference table. This table is printed at the end of each location for every trace specified.

Field Name: QW0037AC
IFCID 038 - Active Write Start

This topic shows detailed information about “Record Trace - IFCID 038 - Active Write Start”.

Record trace - IFCID 038 - Active Write Start

The field labels shown in the following sample layout of “Record Trace - IFCID 038 - Active Write Start” are described in the following section.

<table>
<thead>
<tr>
<th>DSID</th>
<th>ACTLG101</th>
<th>COPY</th>
<th>1</th>
<th>ACE</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>QW0038VR</td>
<td>4345856</td>
<td>QW0038FR</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>QW0038LR</td>
<td>155340800</td>
<td>QW0038LC</td>
<td>155344772</td>
<td></td>
<td></td>
</tr>
<tr>
<td>QW0038LB</td>
<td>X'7F709470'</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

DSID

The data set identifier of the log manager.

Field Name: QW0038DI

COPY

The copy number of the active log data set.

Field Name: QW0038CN

ACE

The relative number of the agent control element address in the ACE cross-reference table. This table is printed at the end of each location for every trace specified.

Field Name: QW0038AC

CI

The number of contiguous control intervals.

Field Name: QW0038CC
IFCID 039 - Active Write End

This topic shows detailed information about “Record Trace - IFCID 039 - Active Write End”.

Record trace - IFCID 039 - Active Write End

The field labels shown in the following sample layout of “Record Trace - IFCID 039 - Active Write End” are described in the following section.

<table>
<thead>
<tr>
<th>DSID</th>
<th>ACTLG102</th>
<th>COPY</th>
<th>1</th>
<th>ACE</th>
<th>17</th>
</tr>
</thead>
<tbody>
<tr>
<td>RET</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>QW0039RC</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**DSID**

The data set identifier.

**Field Name:** QW0039DI

**COPY**

The copy number of the active log data set.

**Field Name:** QW0039CN

**ACE**

The relative number of the agent control element address in the ACE cross-reference table. This table is printed at the end of each location for every trace specified.

**Field Name:** QW0039AC

**RET**

The return code.

**Field Name:** QW0039RT
IFCID 040 - Archive Write Start

This topic shows detailed information about “Record Trace - IFCID 040 - Archive Write Start”.

Record trace - IFCID 040 - Archive Write Start

The field labels shown in the following sample layout of “Record Trace - IFCID 040 - Archive Write Start” are described in the following section.

DSID    DSIDNAME

DSID

The data set identifier of the log manager.

Field Name: QW0040DI
IFCID 041 - Archive Write End

This topic shows detailed information about “Record Trace - IFCID 041 - Archive Write End”.

Record trace - IFCID 041 - Archive Write End

The field labels shown in the following sample layout of “Record Trace - IFCID 041 - Archive Write End” are described in the following section.

RET 0 BLOCKS 5

RET

The return code.

Field Name: QW0041RT

BLOCKS

The number of blocks written.

Field Name: QW0041BW
IFCID 042 - Checkpoint Start

This topic shows detailed information about “Record Trace - IFCID 042 - Checkpoint Start”.

When present, data is printed in dump format, otherwise NO DATA is printed.
IFCID 043 - Checkpoint End

Record trace - IFCID 043 - Checkpoint End

The field labels shown in the following sample layout of “Record Trace - IFCID 043 - Checkpoint End” are described in the following section.

**RBA**

The beginning checkpoint RBA.

**Field Name:** QW0043BC
IFCID 044 - Lock Suspend

This topic shows detailed information about “Record Trace - IFCID 044 - Lock Suspend”.

Record trace - IFCID 044 - Lock Suspend

The field labels shown in the following sample layout of “Record Trace - IFCID 044 - Lock Suspend” are described in the following section.

LOCK RES TYPE

The locked resource type.

Note: For data sharing, SKELETON CURSOR TABLE LOCKING and SKELETON PACKAGE TABLE LOCK are LP-locks (an LP-lock has an L-lock component and a P-lock component).

Field Name: QW0044KT

DBID

The database ID. This field is not applicable if the value in LOCK RES TYPE is:

SKELETON CURSOR TABLE LOCKING
SKELETON PACKAGE TABLE LOCK
 COLLECTION
 ALTER BUFFER POOL

Deduced from the DB2 fields QW0044KD, QW0105DN or QW0107DN.

When present, the database name is shown, otherwise the decimal identifier from QW0044KD is shown, or N/A when this value is 0.

Field Name: RT0044DB

OBID

The object ID of the table space or page set. This field is not applicable if the value in LOCK RES TYPE is:

SKELETON CURSOR TABLE LOCKING
SKELETON PACKAGE TABLE LOCK
 COLLECTION
 ALTER BUFFER POOL

Deduced from the DB2 fields QW0044OB, QW0105TN, QW0107TN, QW0105DB, or QW0107DN.

The table space or object name is shown, when present, otherwise the decimal identifier from QW0044OB is shown, or N/A when this value is 0.

Field Name: RT0044OB

RESOURCE ID

The hexadecimal identifier of the small resource. If LOCK RES TYPE is:

DATA PAGE LOCKING
First 3 bytes are the page number
IFCID 044 - Lock Suspend

DATA SET LOCKING (PARTITION)
Last byte is the partition number

INDEX PAGE LOCKING
First 3 bytes are the page number and the last byte is the subpage number

HASH ANCHOR LOCK
First 3 bytes are the page number and the last byte is the anchor point ID

CS-READ DRAIN
Last byte is the partition number (optional)

RR-READ DRAIN
Last byte is the partition number (optional)

WRITE DRAIN
Last byte is the partition number (optional)

ROW LOCK
First 3 bytes are the page number and the last byte is the row ID of the record

INDEX END OF FILE LOCK
Last byte is the partition number (optional)

Note: In large partitioned table spaces, the page number covers 4 bytes instead of 3.

This field is not applicable if the value in LOCK RES TYPE is SKELETON CURSOR TABLE LOCKING, SKELETON PACKAGE TABLE LOCK, COLLECTION or ALTER BUFFER POOL. If the value is UTILITY Serialization LOCK or BINDLOCK, N/A is printed.

Field Name: QW0044KR

COLL
The collection identifier. This field is only printed if the value in LOCK RES TYPE is SKELETON PACKAGE TABLE LOCKING.

The package identifier. This field is only printed if the value in LOCK RES TYPE is SKELETON PACKAGE TABLE LOCKING.

The consistency token. This field is only printed if the value in LOCK RES TYPE is SKELETON PACKAGE TABLE LOCKING.

The buffer pool ID. This field is only printed if the value in LOCK RES TYPE is ALTER BUFFER POOL.

Field Name: QW0044RN

IRLM FUNC CODE
The IRLM function code.

Field Name: QW0044FC

STATE
The lock state.

Field Name: QW0044ST

DURATION
The lock duration:

**MANUAL** Varies depending on the ISOLATION parameter (QW0044DR=x '20')

**MANUAL+1**
Temporary change of consistency level from CS to RR during bind and DDL (QW0044DR=x '21')

**COMMIT** Until commit (QW0044DR=x '40')

**COMMIT+1**
Past commit; applies to locks needed to maintain the position for a cursor opened WITH HOLD (QW0044DR=x '41')

**ALLOCATION**
Until deallocation (QW0044DR= '60')

**PLAN** For the duration of the plan (QW0044DR=x '80')

**UTIL** For the duration of the utility execution (QW0044DR=x '81')

**INTEREST**
Duration used for P-locks (QW0044DR=x 'FE')

**FREE ALL**
Until all locks are freed (QW0044DR=x 'FF')

**N/A** Not applicable for NOTIFY SUSPEND

Field Name: QW0044DR

**REASON SUSP**
The reason for the suspend:

**LC** IRLM latch contention

**IQ** IRLM queued request

**LR** Local resource contention

**GR** Global resource contention

**IS** Intersystem communication

**N** Notify message sent

**LS** No longer used

**RL** Contention with a retained lock

Field Name: QW0044WS

**REQ TOKEN**
The IRLM lock request token.

Field Name: QW0044RT

**LOCK ATTRIBUTES**
The lock attributes.

Field Name: QW0044FO

**PROP TO XES**
Indicates whether the request was propagated to XES by IRLM.

Field Name: QW0044Y1
IFCID 044 - Lock Suspend

**ASYN TO XES**
Indicates whether the request was sent to XES asynchronously by IRLM.

*Field Name:* QW0044Y2

**PARENT TOKEN**
The parent token for explicit hierarchical locking. This field is valid if the DB2 subsystem is a member of a data sharing group.

*Field Name:* QW0044PT

**LOCK HASH VALUE**
The hash value of the locked resource.

*Field Name:* QW0044LH
IFCID 045 - Lock Resume

This topic shows detailed information about “Record Trace - IFCID 045 - Lock Resume”.

Record trace - IFCID 045 - Lock Resume

The field labels shown in the following sample layout of “Record Trace - IFCID 045 - Lock Resume” are described in the following section.

**REASON FOR RESUME**

The reason for the lock resume.

**Field Name:** QW0045R

**REASON FOR SUSPEND**

The reason for the suspension. The nonserviceability values are:

**IRLM LATCH CONTENTION**

Indicates whether IRLM latch contention occurred. (QW0045W1)

**IRLM QUEUED REQUEST**

Indicates whether there was an IRLM queued request. (QW0045W2)

**LOCAL RESOURCE CONTENTION**

Indicates whether local resource contention occurred. (QW0045W3)

**RETAINED LOCK CONTENTION**

Indicates whether there was contention with a retained lock. (QW0045W4)

**GLOBAL RESOURCE CONTENTION**

Indicates whether intersystem communication was required to resolve an IRLM request. (QW0045W5)

**INTER-SYSTEM MESSAGE SENDING**

Indicates whether any intersystem message was sent. (QW0045W7)

**Field Name:** QW0045SR

**GLOBAL CONTENTION EXTENT**

The extent of global contention. This is applicable if the value in the GLOBAL RESOURCE CONTENTION field is YES. The nonserviceability values are:
IFCID 045 - Lock Resume

**XES GLOBAL CONTENTION**
Indicates whether XES global contention occurred. DB2 field QW0045X3.

**IRLM GLOBAL CONTENTION**
Indicates whether IRLM global contention occurred. DB2 field QW0045X4.

**FALSE CONT OR CONV**
Indicates whether XES global contention or IRLM global contention occurred. DB2 field QW0045X4.

**Field Name:** QW0045XR
This topic shows detailed information about “Record Trace - IFCID 046 - IBM Service Record”.

This record is for IBM service use.
This topic shows detailed information about “Record Trace - IFCID 047 - IBM Service Record”.

This record is for IBM service use.
This topic shows detailed information about “Record Trace - IFCID 048 - IBM Service Record”.

This record is for IBM service use.
IFCID 049 - IBM Service Record

This topic shows detailed information about “Record Trace - IFCID 049 - IBM Service Record”.

This record is for IBM service use.
This topic shows detailed information about “Record Trace - IFCID 050 - IBM Service Record”.

This record is for IBM service use.
This topic shows detailed information about “Record Trace - IFCID 051 - IBM Service Record”.

This record is for IBM service use.
This topic shows detailed information about “Record Trace - IFCID 052 - IBM Service Record”.

This record is for IBM service use.
IFCID 053 - SQL Describe/Commit/Rollback/Remote Statement

This topic shows detailed information about “Record Trace - IFCID 053 - SQL Describe/Commit/Rollback/Remote Statement”.

If this event is not recognized, UNRECOG CMD is printed.

The following data is printed in the DATA column:

Record trace - IFCID 053 - SQL Describe/Commit/Rollback/Remote Statement

The field labels shown in the following sample layout of “Record Trace - IFCID 053 - SQL Describe/Commit/Rollback/Remote Statement” are described in the following section.

LOCATION NAME
The location name.
Field Name: QW0053LN

PKG COLLECTION ID
The package collection identifier.
Field Name: QW0053PC

PROGRAM NAME
The name of the program.
Field Name: QW0053PN

CONSISTENCY TOKEN
The consistency token.
Field Name: QW0053TS

STATEMENT NUMBER
The number of the statement executed.
Field Name: QW0053SN

QUERY COMMAND ID
The ID of the query command.
Field Name: QW0053QID

QUERY INSTANCE ID
The ID of the query instance.
Field Name: QW0053CID

SQL REQUEST TYPE
The type of SQL request.
Field Name: QW0053TOS

EXPANSION REASON
The reason for the expansion. It can have the following values:

GET_ARCHIVE
Expansion caused by built-in SYSIBMADM.GET_ARCHIVE global variable.

TEMPORAL BUSINESS_TIME
Expansion caused by the current temporal BUSINESS_TIME special register.

TEMPORAL SYSTEM_TIME
Expansion caused by the current temporal SYSTEM_TIME special register.

TEMPORAL SYSTEM_TIME & BUSINESS_TIME
Expansion caused by the current temporal SYSTEM_TIME & current temporal BUSINESS_TIME special registers.

NO EXPANSION
The query does not contain any expansion.

Field Name: QW0053ER

SQLCA CONTENTS
This section contains the SQLCA fields. It is only printed if the value in the ENTRY/EXIT TYPE field is RETURNED.

Field Name: QW0053SQ

DATA TYPE
The scan type identification:

INDX   Index scan
SEQD   Sequential data scan
SEQW   Sequential data work-file scan

Field Name: QW0053ID

ROW PROC
The number of rows processed.
Field Name: QW0053RP

ROW EXAM
The number of rows examined. If DATA TYPE shows INDX, this number is the number of index entries (not rows) scanned.

Field Name: QW0053LA

STG1-QUAL
The number of rows qualified at stage 1.
Field Name: QW0053DQ
IFCID 053 - SQL Describe/Commit/Rollback/Remote Statement

**STG2-QUAL**
The number of rows qualified at stage 2.
*Field Name:* QW0053RQ

**ROW INSRT**
The number of rows inserted.
*Field Name:* QW0053IN

**ROW UPDTE**
The number of rows updated.
*Field Name:* QW0053UP

**ROW DELET**
The number of rows deleted. If the delete was a mass delete, the indicator MASS is printed.
*Field Name:* QW0053DE

**PAGES**
The number of get page requests issued by the data manager to the buffer manager. Note that for an index scan the value includes the number of index pages scanned but not the number of index subpages scanned.
*Field Name:* QW0053PS

**RI SCAN**
The number of additional pages scanned for referential integrity.
*Field Name:* QW0053PR

**RI DELET**
The number of additional rows deleted for referential integrity.
*Field Name:* QW0053DR
This topic shows detailed information about “Record Trace - IFCID 055 - Set SQLID”.

This IFCID is written when a SET CURRENT SQLID STATEMENT is issued. It shows the previous SQLID, the new SQLID, and whether the statement succeeded. This record is only written at the application server when DRDA protocol is used.

This record is written when performance trace class 3 is on. This record is written when audit class 7 is on. MONITOR1 privilege is required for reading via ifi.

The SQLID is the SQL authorization ID of the process. This is:

- The authorization ID used for authorization checking on dynamically prepared CREATE, GRANT, and REVOKE SQL statements
- The owner of a table space, database, storage group, or synonym created by a dynamically issued CREATE statement
- The implicit qualifier of all table, view, alias, and index names specified in dynamic SQL statements.

The initial value of CURRENT SQLID can be provided by the connection or sign-on exit routine, otherwise the initial value is the primary authorization ID of the process.

**Record trace - IFCID 055 - Set SQLID**

The field labels shown in the following sample layout of “Record Trace - IFCID 055 - Set SQLID” are described in the following section.

```
SET SQLID  NETWORKID: DEIBMIPS LUNAME: IPSAU851 LUNSEQ: 1
PREV SQLID: DB2PM
NEW SQLID: DB2PM
STATUS : SUCCESSFUL EXECUTION
```

**PREV SQLID**

The initial value of the SQLID before execution of the request.

**Field Name:** QW0055OI

**NEW SQLID**

If the command completed successfully, the new value of the SQLID is shown. If the command did not complete successfully, the value of the attempted SQLID change is shown.

**Field Name:** QW0055NI

**STATUS**

The success or failure of the attempted authorization change. Possible values are:

- SUCCESS or SUCCESSFUL for a successful authorization change
- FAILURE for a failed attempt

**Note:** The SQL statement is always successful if the user has SYSADM authority.

**Field Name:** QW0055ST
IFCID 056 - IBM Service Record

This topic shows detailed information about “Record Trace - IFCID 056 - IBM Service Record”.

This record is for IBM service use.

Record trace - IFCID 056 - IBM Service Record

The field labels shown in the following sample layout of “Record Trace - IFCID 056 - IBM Service Record” are described in the following section.

QW0056LA: X'1122334455667788'  QW0056LC: X'E2'
QW0056LF: X'00000A00'

QW0056LA

This field is for IBM service use.

Field Name: QW0056LA

QW0056LC

This field is for IBM service use.

Field Name: QW0056LC

QW0056LF

This field is for IBM service use.

Field Name: QW0056LF
This topic shows detailed information about “Record Trace - IFCID 057 - IBM Service Record”.

This record is for IBM service use.

**Record trace - IFCID 057 - IBM Service Record**

The field labels shown in the following sample layout of “Record Trace - IFCID 057 - IBM Service Record” are described in the following section.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>QW0057LA</td>
<td>X'1123345667788'</td>
</tr>
<tr>
<td>QW0057LC</td>
<td>X'88'</td>
</tr>
<tr>
<td>QW0057TS</td>
<td>X'00000A00'</td>
</tr>
</tbody>
</table>

**QW0057LA**

This field is for IBM service use.

**Field Name:** QW0057LA

**QW0057LC**

This field is for IBM service use.

**Field Name:** QW0057LC

**QW0057TS**

This field is for IBM service use.

**Field Name:** QW0057TS
This topic shows detailed information about “Record Trace - IFCID 058 - End SQL”.

**Record trace - IFCID 058 - End SQL**

The field labels shown in the following sample layout of “Record Trace - IFCID 058 - End SQL” are described in the following section.

| LOCATION NAME | Field Name: QW0058LN |
| PKG COLLECTION ID | Field Name: QW0058PC |
| PROGRAM NAME | Field Name: QW0058PN |
| CONSISTENCY TOKEN | Field Name: QW0058TS |
| STATEMENT NUMBER | Field Name: QW0058SN |
| QUERY COMMAND ID | Field Name: QW0058CID |
| QUERY INSTANCE ID | Field Name: QW0058QID |
| SQL REQUEST TYPE | Field Name: QW0058TOS |
EXPANSION REASON

The reason for the expansion. It can have the following values:

GET_ARCHIVE
Expansion caused by built-in SYSIBMADM.GET_ARCHIVE global variable.

TEMPORAL BUSINESS_TIME
Expansion caused by the current temporal BUSINESS_TIME special register.

TEMPORAL SYSTEM_TIME
Expansion caused by the current temporal SYSTEM_TIME special register.

TEMPORAL SYSTEM_TIME & BUSINESS_TIME
Expansion caused by the current temporal SYSTEM_TIME & current temporal BUSINESS_TIME special registers.

NO EXPANSION
The query does not contain any expansion.

Field Name: QW0058EXR

SQLCA CONTENTS
This section contains the SQLCA fields. It is only printed if the value in the ENTRY/EXIT TYPE field is RETURNED.

Field Name: QW0058SQ

DATA TYPE
The scan type identification.

INDX  Index scan
SEQD  Sequential data scan
SEQW  Sequential data work-file scan

Field Name: QW0058ID

ROW PROC
The number of rows processed.

Field Name: QW0058RP

ROW EXAM
The number of rows examined. If DATA TYPE shows INDX, this number is the number of index entries (not rows) scanned.

Field Name: QW0058LA

STG1-QUAL
The number of rows qualified at stage 1.

Field Name: QW0058DQ

STG2-QUAL
The number of rows qualified at stage 2.

Field Name: QW0058RQ
IFCID 058 - End SQL

ROW INSRT
The number of rows inserted.

Field Name: QW0058IN

ROW UPDTE
The number of rows updated.

Field Name: QW0058UP

ROW DELET
The number of rows deleted. If the delete was a mass delete, the indicator MASS is printed.

Field Name: QW0058DE

PAGES
The number of get page requests issued by the data manager to the buffer manager. Note that for an index scan, the value includes the number of index pages scanned but not the number of index subpages scanned.

Field Name: QW0058PS

RI SCAN
The number of additional pages scanned for referential integrity.

Field Name: QW0058PR

RI DELET
The number of additional rows deleted for referential integrity.

Field Name: QW0058DR

LOB SCAN
Additional pages scanned in a LOB table space.

Field Name: QW0058PL

LOB UPDTE
Number of LOB data pages updated by SQL INSERT or SQL UPDATE.

Field Name: QW0058UL
IFCID 059 - Fetch Start

This topic shows detailed information about “Record Trace - IFCID 059 - Fetch Start”.

Record trace - IFCID 059 - Fetch Start

The field labels shown in the following sample layout of “Record Trace - IFCID 059 - Fetch Start” are described in the following section.

START LOCATION NAME : PM05D851
PKG COLLECTION ID: HSRTEP2VL1XXXXXXXXX2XXXXXXXXX3XXXXXXX4XXXXXXXXX5XXXXXXXXX6XXXXXXXXX7XXXXXXXXX8XXXXXXXXX9XXXXXXXXX0XX
XXXXXXX1XXXXXXXXX2XXXXXXXZ
PROGRAM NAME : DSN0EP2L
CURSOR NAME : C1
CONSISTENCY TOKEN: X'172A1C98193C380E'
STATEMENT NUMBER : 1627
STATEMENT TYPE : X'01'
QUERY COMMAND ID : ----
QUERY INSTANCE ID: ----
FETCH SENSITIVITY: UNSPECIFIED
FETCH ORIENTATION: NEXT

LOCATION NAME
The location name.
Field Name: QW0059LN

PKG COLLECTION ID
The package collection identifier.
Field Name: QW0059PC

PROGRAM NAME
The program name.
Field Name: QW0059PN

CURSOR NAME
The name of the cursor used by the FETCH statement.
Field Name: QW0059CN

CONSISTENCY TOKEN
The consistency token.
Field Name: QW0059TS

STATEMENT NUMBER
The statement number.
Field Name: QW0059SN

STATEMENT TYPE
The statement type. X'01' indicates FETCH.
Field Name: QW0059ST

QUERY COMMAND ID
The ID of the query command.
IFCID 059 - Fetch Start

Field Name: QW0059CID

QUERY INSTANCE ID
The ID of the query instance.

Field Name: QW0059QID

FETCH SENSITIVITY
Identifies the fetch sensitivity. It can be one of the following:
• Sensitive
• Insensitive
• Unspecified

Field Name: QW0059FS

FETCH ORIENTATION
Identifies the fetch orientation. It can be one of the following:
• First
• Last
• Before
• After
• Next
• Previous
• Current
• Absolute
• Relative
• Unspecified

Field Name: QW0059FO
IFCID 060 - Select Start

This topic shows detailed information about “Record Trace - IFCID 060 - Select Start”.

Record trace - IFCID 060 - Select Start

The field labels shown in the following sample layout of “Record Trace - IFCID 060 - Select Start” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOCATION NAME</td>
<td>The location name.</td>
</tr>
<tr>
<td>Field Name: QW0060LN</td>
<td></td>
</tr>
<tr>
<td>PKG COLLECTION ID</td>
<td>The package collection identifier.</td>
</tr>
<tr>
<td>Field Name: QW0060PC</td>
<td></td>
</tr>
<tr>
<td>PROGRAM NAME</td>
<td>The program name.</td>
</tr>
<tr>
<td>Field Name: QW0060PN</td>
<td></td>
</tr>
<tr>
<td>CONSISTENCY TOKEN</td>
<td>The consistency token.</td>
</tr>
<tr>
<td>Field Name: QW0060TS</td>
<td></td>
</tr>
<tr>
<td>STATEMENT NUMBER</td>
<td>The statement number of the statement executed.</td>
</tr>
<tr>
<td>Field Name: QW0060SN</td>
<td></td>
</tr>
<tr>
<td>STATEMENT TYPE</td>
<td>The statement type. X'00' indicates SELECT.</td>
</tr>
<tr>
<td>Field Name: QW0060ST</td>
<td></td>
</tr>
<tr>
<td>QUERY COMMAND ID</td>
<td>The ID of the query command.</td>
</tr>
<tr>
<td>Field Name: QW0060CID</td>
<td></td>
</tr>
<tr>
<td>QUERY INSTANCE ID</td>
<td>The ID of the query instance.</td>
</tr>
<tr>
<td>Field Name: QW0060QID</td>
<td></td>
</tr>
<tr>
<td>ISOLATION</td>
<td></td>
</tr>
</tbody>
</table>

LOCATION NAME

The location name.

Field Name: QW0060LN

PKG COLLECTION ID

The package collection identifier.

Field Name: QW0060PC

PROGRAM NAME

The program name.

Field Name: QW0060PN

CONSISTENCY TOKEN

The consistency token.

Field Name: QW0060TS

STATEMENT NUMBER

The statement number of the statement executed.

Field Name: QW0060SN

STATEMENT TYPE

The statement type. X'00' indicates SELECT.

Field Name: QW0060ST

QUERY COMMAND ID

The ID of the query command.

Field Name: QW0060CID

QUERY INSTANCE ID

The ID of the query instance.

Field Name: QW0060QID

ISOLATION
The isolation level:

**RR**  Repeatable read

**CS**  Cursor stability

**RS**  Read stability

**UR**  Uncommitted read

**Field Name:** QW0060I

**REOPTIMIZATION**

Indicates whether the access path of the SQL statement was reoptimized at run time.

**Field Name:** QW0060RO
IFCID 061 - Insert/Update/Delete Start

This topic shows detailed information about “Record Trace - IFCID 061 - Insert/Update/Delete Start”.

Record trace - IFCID 061 - Insert/Update/Delete Start

The field labels shown in the following sample layout of “Record Trace - IFCID 061 - Insert/Update/Delete Start” are described in the following section.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>START LOCATION NAME</td>
<td>PM05D851</td>
</tr>
<tr>
<td>PKG COLLECTION ID</td>
<td>DB2PM</td>
</tr>
<tr>
<td>PROGRAM NAME</td>
<td>DO009PC4</td>
</tr>
<tr>
<td>CONSISTENCY TOKEN</td>
<td>X'1746B2741FC9F534'</td>
</tr>
<tr>
<td>STATEMENT NUMBER</td>
<td>168</td>
</tr>
<tr>
<td>STATEMENT TYPE</td>
<td>UPDATE TYPE- NON CURSOR</td>
</tr>
<tr>
<td>CURSOR NAME</td>
<td>N/P</td>
</tr>
<tr>
<td>QUERY COMMAND ID</td>
<td>N/P</td>
</tr>
<tr>
<td>QUERY INSTANCE ID</td>
<td>N/P</td>
</tr>
<tr>
<td>ISOLATION</td>
<td>RR</td>
</tr>
<tr>
<td>REOPTIMIZATION</td>
<td>NO</td>
</tr>
</tbody>
</table>

LOCATION NAME

The location name.

Field Name: QW0061LN

PKG COLLECTION ID

The package collection ID.

Field Name: QW0061PC

PROGRAM NAME

The program name.

Field Name: QW0061PN

CONSISTENCY TOKEN

The consistency token.

Field Name: QW0061TS

STATEMENT NUMBER

The statement number of the statement executed.

Field Name: QW0061SN

STATEMENT TYPE

The statement type. Possible values are:

- STATEMENT TYPE
- INSERT TYPE
- UPDATE TYPE- NON CURSOR
- UPDATE TYPE- CURSOR
- MERGE TYPE
- DELETE TYPE- NON CURSOR
- DELETE TYPE- CURSOR
- TRUNCATE TYPE

Field Name: QW0061ST
IFCID 061 - Insert/Update/Delete Start

CURSOR NAME
The name of the cursor.
Field Name: QW0061CN

QUERY COMMAND ID
The ID of the query command.
Field Name: QW0061CI

QUERY INSTANCE ID
The ID of the query instance.
Field Name: QW0061QI

ISOLATION
The isolation level:
RR   Repeatable read
CS   Cursor stability
RS   Read stability
Field Name: QW0061I

REOPTIMIZATION
Indicates whether the access path of the SQL statement was reoptimized at run time.
Field Name: QW0061RO
IFCID 062 - DDL Start

This topic shows detailed information about “Record Trace - IFCID 062 - DDL Start”.

Record trace - IFCID 062 - DDL Start

The field labels shown in the following sample layout of “Record Trace - IFCID 062 - DDL Start” are described in the following section.

```
START STATEMENT TYPE: ALTER VIEW
OBJECT TYPE   : TABLESPACE
OBJECT NAME   : HSRPDSBU
```

**STATEMENT TYPE**

The type of statement that is processed. Possible values are:

- ALTER JAR
- ALTER MASK
- ALTER PERMISSION
- ALTER SEQUENCES
- ALTER TRUSTED CONTEXT
- CREATE JAR
- CREATE MASK
- CREATE PERMISSION
- CREATE ROLE
- CREATE SEQUENCES
- CREATE TRUSTED CONTEXT
- DROP JAR
- DROP MASK
- DROP PERMISSION
- DROP ROLE
- DROP SEQUENCES
- DROP TRUSTED CONTEXT
- RENAME INDEX

**Field Name:** QW0062ST

**OBJECT TYPE**

The type of object that is processed. Possible values are:

- COLUMN MASK
- JAR
- SEQUENCES
- TRUSTED CONTEXT
- ROLE
- ROW PERMISSION

**Field Name:** QW0062OT

**OBJECT NAME**

The object name. The name does not include high-level qualifiers.

**Field Name:** QW0062ON
IFCID 063 - SQL Statement

This topic shows detailed information about “Record Trace - IFCID 063 - SQL Statement”.

Record trace - IFCID 063 - SQL Statement

The field labels shown in the following sample layout of “Record Trace - IFCID 063 - SQL Statement” are described in the following section.

SQL STATEMENT NETWORKID: DEIBMIPS LUNAME: IPSAUSB1 LUWSEQ: 1 OPTIONS: X'04' HOST LANG: N/A SQL STATEMENT:

This field consists of 8 bits. The bits indicate the parser options and the host language. The four most significant bits describe the parser options:

Bit 7  String delimiter (0 = apostrophe; 1 = quotation mark)
Bit 6  Decimal separator (0 = period; 1 = comma)
Bit 5  SQL delimiter (0 = apostrophe; 1 = quotation mark)
Bit 4  Mixed data (0 = no; 1 = yes)
Bit 3  Reserved
Bits 0 to 2  Host language

The three least significant bits (0 to 2) are the host language bit mask. The hexadecimal value indicates the host language:

001  Host language is Assembler
010  Host language is COBOL
011  Host language is PL/I
100  Host language is Dynamic SQL
101  Host language is FORTRAN
110  Host language is COBOL2
Look at HOST LANG field

Field Name: QW0063OT

HOST LANG

Additional host language option. This field is optional. When the OPTIONS host language bit mask is x'7', it indicates the host language.

Field Name: QW0063HL

SQL STATEMENT

The SQL statement being processed.

Note:

- SQL text longer than 5000 characters is truncated.
- Host variables are represented as :H

Field Name: QW0063ST

STATEMENT IDENTIFIER

The type of statement.

Field Name: QW0063TY

TYPE OF STATEMENT

The statement identifier.

Field Name: QW0063SI

CCSID

The coded character set identifier (CCSID).

Field Name: QW0063CC
IFCID 064 - Prepare Start

This topic shows detailed information about “Record Trace - IFCID 064 - Prepare Start”.

Record trace - IFCID 064 - Prepare Start

The field labels shown in the following sample layout of “Record Trace - IFCID 064 - Prepare Start” are described in the following section.

START LOCATION NAME : PM05DB51
PKG COLLECTION ID: HSRTEP2VL1XXXXXXXX2XXXXXXXXX3XXXX
XXXXXXXX4XXXXXXXX5XXXXXXXX6XXXXXXXX
X7XXXXXXXX8XXXXXXXX9XXXXXXXX0XX
XXXXXXXX1XXXXXXXX2XXXXXXXXZ
PROGRAM NAME : DSN8EP2L
TIME STAMP : X'172A1C98193C380E'
STATEMENT NUMBER : 1550
STATEMENT TYPE : X'81'
CURSOR NAME : C1
QUERY COMMAND ID : QRYCMD01
QUERY INSTANCE ID: QRYINS01

LOCATION NAME
The location name.
Field Name: QW0064LN

PKG COLLECTION ID
The package collection identifier.
Field Name: QW0064CI

PROGRAM NAME
The program name.
Field Name: QW0064PN

TIME STAMP
The hexadecimal value of the precompiler timestamp.
Field Name: QW0064TS

STATEMENT NUMBER
The statement number.
Field Name: QW0064SN

STATEMENT TYPE
The statement types:
'81' Prepare a cursor section.
'80' Prepare a noncursor section.
'C1' Implicit prepare of a cursor section.
'C0' Implicit prepare of a noncursor section.
Field Name: QW0064ST

CURSOR NAME
The name of the cursor used by the PREPARE statement.
IFCID 064 - Prepare Start

**Field Name:** QW0064CN

**QUERY COMMAND ID**
- The ID of the query command.

**Field Name:** QW0064CID

**QUERY INSTANCE ID**
- The ID of the query instance.

**Field Name:** QW0064QID
IFCID 065 - Open Cursor

Record trace - IFCID 065 - Open Cursor

The field labels shown in the following sample layout of “Record Trace - IFCID 065 - Open Cursor” are described in the following section.

LOCATION NAME

The location name.

Field Name: QW0065LN

PKG COLLECTION ID

The package collection identifier.

Field Name: QW0065PC

PROGRAM NAME

The program name.

Field Name: QW0065PN

CONSISTENCY TOKEN

The consistency token.

Field Name: QW0065TS

STATEMENT NUMBER

The statement number.

Field Name: QW0065SN

STATEMENT TYPE

The statement type. X’91’ indicates OPEN.

Field Name: QW0065ST

CURSOR NAME

The name of the cursor used by the OPEN cursor statement.
Field Name: QW0065CN

QUERY COMMAND ID
The ID of the query command.
Field Name: QW0065CID

QUERY INSTANCE ID
The ID of the query instance.
Field Name: QW0065QID

ISOLATION
The isolation level:
RR  Repeatable read
CS  Cursor stability
RS  Read stability
UR  Uncommitted read
XR  Repeatable read with X lock
XS  Read stability with X lock
Field Name: QW0065I

REOPTIMIZATION
Indicates whether the access path of the SQL statement was reoptimized at run time.
Field Name: QW0065RO

CURSOR SCROLLABILITY
Identifies the cursor scrollability. It can be one of the following:
• Scroll
• None-scroll
Field Name: QW0065SC

CURSOR SENSITIVITY
Identifies the cursor sensitivity. It can be one of the following:
• Sensitive
• Inensitive
• Unspecified
Field Name: QW0065SV

CUR RESULT TABLE TYPE
Identifies the type of the cursor result table. It can be one of the following:
• Static
• Dynamic
• Unspecified
Field Name: QW0065RT

CURSOR CLOSE COMMIT
IFCID 065 - Open Cursor

The cursor attribute implicit commit, which closed the cursor. Possible values are:
• IMPLICIT COMMIT
• NO IMPLICIT COMMIT
• N/A

Otherwise the values are shown in hexadecimal.

Field Name: QW0065TY
IFCID 066 - Close Cursor

This topic shows detailed information about “Record Trace - IFCID 066 - Close Cursor”.

Record trace - IFCID 066 - Close Cursor

The field labels shown in the following sample layout of “Record Trace - IFCID 066 - Close Cursor” are described in the following section.

CURSOR LOCATION NAME : PM05D851
PKG COLLECTION ID : HSRTEP2VL1XXXXXXXXX2XXXXXXXXX3
XXXXXXXXXXXXXXXXXXX2XXXXXXXXX3
XXXXXXXXXXXXXXXXXXX2XXXXXXXXX3
XXXXXXXXXXXXXXXXXXX2XXXXXXXXX3
XXXXXXXXXXXXXXXXXXX2XXXXXXXXX3
XXXXXXXZ
PROGRAM NAME : DSN@EP2L
CONSISTENCY TOKEN : X'172A1C98193C380E'
STATEMENT NUMBER : 1889
STATEMENT TYPE : X'A1'
CURSOR NAME : C1
CLOSE STMT TYPE : IMPLICIT
QUERY COMMAND ID : ----
QUERY INSTANCE ID : ----

LOCATION NAME
The location name.

Field Name: QW0066LN

PKG COLLECTION ID
The package collection identifier.

Field Name: QW0066PC

PROGRAM NAME
The program name.

Field Name: QW0066PN

CONSISTENCY TOKEN
The consistency token.

Field Name: QW0066TS

STATEMENT NUMBER
The statement number.

Field Name: QW0066SN

STATEMENT TYPE
The statement type. X'A1' indicates CLOSE.

Field Name: QW0066ST

CURSOR NAME
The name of the cursor used by the CLOSE cursor statement.

Field Name: QW0066CN

CLOSE STMT TYPE
The Close statement type. Possible values are:
IFCID 066 - Close Cursor

- IMPLICIT
- EXPLICIT
- N/A
  Otherwise the values are shown in hexadecimal.

**Field Name:** QW0066TY

**QUERY COMMAND ID**

The ID of the query command.

**Field Name:** QW0066CID

**QUERY INSTANCE ID**

The ID of the query instance.

**Field Name:** QW0066QID
IFCID 067 - Accounting

This topic shows detailed information about “Record Trace - IFCID 067 - Accounting”.

When present, data is printed in dump format, otherwise NO DATA is printed.
IFCID 068 - Rollback Start

Record trace - IFCID 068 - Rollback Start

The field labels shown in the following sample layout of “Record Trace - IFCID 068 - Rollback Start” are described in the following section.

**Field Name:** QW0068CK

**PSWKEY**

The PSW key of the holder.
This topic shows detailed information about “Record Trace - IFCID 069 - IBM Service Record”.

This record is for IBM service use.
IFCID 070 - Commit Phase 2 Start

IFCID 070 - Commit Phase 2 Start

This topic shows detailed information about “Record Trace - IFCID 070 - Commit Phase 2 Start”.

Record trace - IFCID 070 - Commit Phase 2 Start

The field labels shown in the following sample layout of “Record Trace - IFCID 070 - Commit Phase 2 Start” are described in the following section.

PSWKEY X’OE’
QW0070FR X’00000000’

PSWKEY

The PSW key of the caller.

Field Name: QW0070CK
This topic shows detailed information about “Record Trace - IFCID 071 - IBM Service Record”.

This record is for IBM service use.
IFCID 072 - Create Thread Start

This topic shows detailed information about “Record Trace - IFCID 072 - Create Thread Start”.

Record trace - IFCID 072 - Create Thread Start

The field labels shown in the following sample layout of “Record Trace - IFCID 072 - Create Thread Start” are described in the following section.

RESOURCE NAME: ABE5B03

RESOURCE NAME

The plan name used in thread creation. If the thread is created to process a DB2 command, the field shows 'BLANK'.

Field Name: QW0072RN
IFCID 073 - Create Thread End

This topic shows detailed information about “Record Trace - IFCID 073 - Create Thread End”.

Record trace - IFCID 073 - Create Thread End

The field labels shown in the following sample layout of “Record Trace - IFCID 073 - Create Thread End” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>QLGTH</td>
<td>The queue length of the create thread request.</td>
</tr>
</tbody>
</table>

Field Name: QW0073QL
IFCID 074 - Terminate Thread Start

This topic shows detailed information about “Record Trace - IFCID 074 - Terminate Thread Start”.

When data is present, it is printed in dump format.
IFCID 075 - Terminate Thread End

This topic shows detailed information about “Record Trace - IFCID 075 - Terminate Thread End”.

Record trace - IFCID 075 - Terminate Thread End

The field labels shown in the following sample layout of “Record Trace - IFCID 075 - Terminate Thread End” are described in the following section.

QW0075RT 0 QW0075RS 0
QW0075CO X'E2E8D5C3'
IFCID 076 - End of Memory Start

Record trace - IFCID 076 - End of Memory Start

This topic shows detailed information about “Record Trace - IFCID 076 - End of Memory Start”.

The field labels shown in the following sample layout of “Record Trace - IFCID 076 - End of Memory Start” are described in the following section.

```
PSWKEY  X'01'  FLGS  X'07'  ASID  256
QW0076SS  X'01234567'  QW0076AM  X'89ABCDEF'
QW0076FC  2  QW0076AS  X'12345678'
```

**PSWKEY**

The PSW key of the SSI caller.

**Field Name:** QW0076CK

**FLGS**

SSI caller flags:

- **X'80'** SSI caller problem state (P-bit)
- **X'40'** A-bit SSI caller AMODE 31 (A-bit)
- **X'20'** Abnormal end of memory

**Field Name:** QW0076F1

**ASID**

The identifier of the end of memory address space.

**Field Name:** QW0076ID
IFCID 077 - End of Memory End

IFCID 077 - End of Memory End

This topic shows detailed information about “Record Trace - IFCID 077 - End of Memory End”.

Record trace - IFCID 077 - End of Memory End

The field labels shown in the following sample layout of “Record Trace - IFCID 077 - End of Memory End” are described in the following section.

RETURN 0 PROCESSED END OF MEMORY?: YES

The return code. This field is always 0.

Field Name: QW0077R0

PROCESSED END OF MEMORY?

Indicates whether end of memory was processed.

Field Name: QW0077PR
IFCID 078 - End of Task Start

This topic shows detailed information about “Record Trace - IFCID 078 - End of Task Start”.

Record trace - IFCID 078 - End of Task Start

The field labels shown in the following sample layout of “Record Trace - IFCID 078 - End of Task Start” are described in the following section.

ACE 2
QW0078AS X'0089F32B' QW0078AG X'008226C8'

ACE

The relative number of the agent control element address in the ACE cross-reference table. This table is printed at the end of each location for every trace specified.

Field Name: QW0078AC
IFCID 079 - End of Task End

This topic shows detailed information about “Record Trace - IFCID 079 - End of Task End”.

When present, data is printed in dump format, otherwise NO DATA is printed.
This topic shows detailed information about “Record Trace - IFCID 080 - IBM Service Record”.

This record is for IBM service use.
This topic shows detailed information about “Record Trace - IFCID 081 - IBM Service Record”.

This record is for IBM service use.
IFCID 082 - Identify Start

Record trace - IFCID 082 - Identify Start

The field labels shown in the following sample layout of “Record Trace - IFCID 082 - Identify Start” are described in the following section.

**PSWKEY**

- The PSW key of the SSI caller.
- Field Name: QW0082CK

**FLAGS**

- The flags of the SSI caller:
  - X'80'  - P-bit of the SSI caller (problem state)
  - X'40'  - A-bit of the SSI caller (AMODE 31)
- Field Name: QW0082F1
IFCID 083 - Identify End

This topic shows detailed information about “Record Trace - IFCID 083 - Identify End”.

Record trace - IFCID 083 - Identify End

The field labels shown in the following sample layout of “Record Trace - IFCID 083 - Identify End” are described in the following section.

END RECOPT: 'BLANK' ACCESS: SUCCESSFUL
CURR SQLID : HSR
ORIG AUTHID: HSR
SECONDARY AUTHORIZATION IDS:
DE#01892 DE#03704 D01DD PMDEV PMDEVX
PMDEVX5 PMDEVX6 PMDEVX7 PMDEV5 PMDEV6
PMDEV61 PMDEV7 PMDEV71 PMDEV72 PMDEV81
PM3704 USERS
ACEE UTOKEN : 'BLANK'
QW0083RT 0 QW0083RS 0
QW0083CT X'C2C1E3C3C8404040'

RECOPT

The record coordination option specification.

Field Name: QW0083RO

ACCESS

The success or failure of the attempted authorization change. Possible values are:

SUCCESSFUL
The access is permitted.

DENIED BY IDENTIFY AUTH EXIT
The access is denied by the authorization exit.

DENIED BY SAF/SECURITY SYSTEM
The access is denied by the security authorization facility or security system.

Field Name: QW0083AD

CURR SQLID

The current SQL authorization ID.

Field Name: QW0083QD

ORIG AUTHID

The original primary authorization ID.

Field Name: QW0083OP

SECONDARY AUTHORIZATION IDS

A list of the secondary authorization IDs. This list is only produced if there are secondary authorization IDs.

Field Name: QW0083SA

ACEE UTOKEN

The ACEE UTOKEN.

Field Name: QW0083UT
IFCID 084 - Prepare Start

This topic shows detailed information about “Record Trace - IFCID 084 - Prepare Start”.

Record trace - IFCID 084 - Prepare Start

The field labels shown in the following sample layout of “Record Trace - IFCID 084 - Prepare Start” are described in the following section.

PSWKEY X’00’
QW0084FR X’00000000’

PSWKEY

The PSW key of the caller.

Field Name: QW0084CK
IFCID 085 - Prepare End

This topic shows detailed information about “Record Trace - IFCID 085 - Prepare End”.

Record trace - IFCID 085 - Prepare End

The field labels shown in the following sample layout of “Record Trace - IFCID 085 - Prepare End” are described in the following section.
This topic shows detailed information about “Record Trace - IFCID 086 - Signon Start”.

**Record trace - IFCID 086 - Signon Start**

The field labels shown in the following sample layout of “Record Trace - IFCID 086 - Signon Start” are described in the following section.

<table>
<thead>
<tr>
<th>PSWKEY</th>
<th>X'00'</th>
</tr>
</thead>
<tbody>
<tr>
<td>QW0086FR</td>
<td>X'00000000'</td>
</tr>
</tbody>
</table>

**PSWKEY**

The PSW key of the SSI caller.

**Field Name:** QW0086CK
IFCID 087 - Signon End

This topic shows detailed information about “Record Trace - IFCID 087 - Signon End”.

Record trace - IFCID 087 - Signon End

The field labels shown in the following sample layout of “Record Trace - IFCID 087 - Signon End” are described in the following section.

ACCESS: SUCCESSFUL
CURRENT SQLID : DEA
ORIGINAL AUTHID: DEA
SECONDARY AUTHORIZATION IDS:
AUTHID02 AUTHID03 AUTHID04
ACEE UTOKEN : 'BLANK'

ACCESS
The success or failure of the attempted access. Possible values are:

SUCCESSFUL
If the access is permitted.

DENIED BY SIGNON AUTH EXIT
If the access was denied by the signon authorization exit.

Field Name: QW0087AD

CURRENT SQLID
The value of the authorization ID as set by the IDENTIFY or SIGNON exit.

Field Name: QW0087QD

ORIGINAL AUTHID
The original value of the authorization ID, as passed to the IDENTIFY or SIGNON authorization exit.

Field Name: QW0087OP

SECONDARY AUTHORIZATION IDS
Lists the secondary authorization IDs set by the IDENTIFY or SIGNON authorization exits. If no secondary authorization IDs exist, this line is not printed. Secondary authorization IDs are printed in rows of five, up to a maximum of 49 rows (245 AUTHIDs).

Field Name: QW0087SA

ACEE UTOKEN
The ACEE UTOKEN.

Field Name: QW0087UT
IFCID 088 - Synch Start

This topic shows detailed information about “Record Trace - IFCID 088 - Synch Start”.

Record trace - IFCID 088 - Synch Start

The field labels shown in the following sample layout of “Record Trace - IFCID 088 - Synch Start” are described in the following section.

PSWKEY  X’00’
QW0088FR  X’00000000’

PSWKEY

The PSW key of the caller.

Field Name: QW0088CK
IFCID 089 - Synch End

This topic shows detailed information about “Record Trace - IFCID 089 - Synch End”.

Record trace - IFCID 089 - Synch End

The field labels shown in the following sample layout of “Record Trace - IFCID 089 - Synch End” are described in the following section.

QW0089FR  X’007BC428’  QW0089RT  0
QW0089RS  0  QW0089RS
IFCID 090 - DB2 Command Start

This topic shows detailed information about “Record Trace - IFCID 090 - DB2 Command Start”.

Record trace - IFCID 090 - DB2 Command Start

The field labels shown in the following sample layout of “Record Trace - IFCID 090 - DB2 Command Start” are described in the following section.

COMMAND: -ARCHIVE LOG MODE(QUIESCE) TIME(2)
PHB X'02BC1040'
COM

The command text.
The input PHB token is extracted from the first 4 bytes of the COMMAND field.

Field Name: QW0090CT
IFCID 091 - Command End

This topic shows detailed information about “Record Trace - IFCID 091 - Command End”.

Record trace - IFCID 091 - Command End

The field labels shown in the following sample layout of “Record Trace - IFCID 091 - Command End” are described in the following section.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>QW0091RC</td>
<td>0</td>
</tr>
<tr>
<td>QW0091RS</td>
<td>0</td>
</tr>
<tr>
<td>QW0091BA</td>
<td>X'7F4B9F10'</td>
</tr>
</tbody>
</table>
IFCID 092 - AMS Command Start

This topic shows detailed information about “Record Trace - IFCID 092 - AMS Command Start”.

Record trace - IFCID 092 - AMS Command Start

The field labels shown in the following sample layout of “Record Trace - IFCID 092 - AMS Command Start” are described in the following section.

COMMAND: DEFINE CLUSTER
(NAME(DSN220C.DSNDBC.CDDB.EMPLOYEE.I0001.A001)
) NOERASE LINEAR OWNER(DB2ADM ) REUSE ) DATA
(NAME(DSN220C.DSNDBD.CDDB.EMPLOYEE.I0001.A001)
) RECORDS( 00000003 00000003) OWNER(DB2ADM )
SHAREOPTIONS(3,3) REUSE VOLUMES('ELURU2'));

COMMAND

The command text.

Field Name: QW0092P1
This topic shows detailed information about “Record Trace - IFCID 093 - IBM Service Record”.

This record is for IBM service.
This topic shows detailed information about “Record Trace - IFCID 094 - IBM Service Record”.

This record is for IBM service use.
This topic shows detailed information about “Record Trace - IFCID 095 - Sort Start”.

When present, data is shown in dump format, otherwise *NO DATA* is printed.
IFCID 096 - Sort End

This topic shows detailed information about “Record Trace - IFCID 096 - Sort End”.

Record trace - IFCID 096 - Sort End

The field labels shown in the following sample layout of “Record Trace - IFCID 096 - Sort End” are described in the following section.

<table>
<thead>
<tr>
<th>COLLECTION ID: DB2PM</th>
</tr>
</thead>
<tbody>
<tr>
<td>PROGRAM NAME : DGO@PC2</td>
</tr>
<tr>
<td>RECNO : 26</td>
</tr>
<tr>
<td>AREA : 53</td>
</tr>
<tr>
<td>KEYSZ : 4</td>
</tr>
<tr>
<td>SIZE : 57</td>
</tr>
<tr>
<td>WORK : 1</td>
</tr>
<tr>
<td>RET : 0</td>
</tr>
<tr>
<td>IWORK : 1</td>
</tr>
<tr>
<td>ROW DEL : 0</td>
</tr>
<tr>
<td>PASSES : 0</td>
</tr>
<tr>
<td>SORT KEYS : 1</td>
</tr>
<tr>
<td>STMMNO : 780</td>
</tr>
<tr>
<td>WORKFILES : 0</td>
</tr>
<tr>
<td>SORT COLUMNS : 7</td>
</tr>
<tr>
<td>PARTITIONING BY SORT : NO</td>
</tr>
<tr>
<td>SORT IN ADDITION : NO</td>
</tr>
<tr>
<td>TYPE : ESA</td>
</tr>
<tr>
<td>PARTITIONING OCCURRED : NOT PARTITIONING</td>
</tr>
<tr>
<td>QW0096IN (S) : 0</td>
</tr>
<tr>
<td>QW0096RD (S) : 0</td>
</tr>
<tr>
<td>QW0096RU (S) : 0</td>
</tr>
</tbody>
</table>

COLLECTION ID

The package collection ID for the query that invokes sort.

Field Name: QW0096PC

PROGRAM NAME

The program name for the query that invokes sort.

Field Name: QW0096PN

RECNO

The number of records sorted.

Field Name: QW0096NR

AREA

The sort data area size in bytes.

Field Name: QW0096DL

KEYSZ

The sort key size in bytes.

Field Name: QW0096KL

SIZE

The sort record size in bytes (the sort key size and the data area size).

Field Name: QW0096WR

WORK

The number of work files used for both input and merge phases.

Field Name: QW0096WF

RET

The sort return code:
0 Successful
4   Empty - sort successful
8   Resource unavailable
12  Sort key too long
16  Error detected by fetch routine during input phase
20  Serious processing error

Field Name: QW0096RC

IWORK

The number of initial work files. The sorting of records can take more than
one work file. The number of work files needed depends on the
distribution of sort key values. The maximum number of work files is
limited by the buffer pool size.

Field Name: QW0096IR

ROW DEL

The number of rows deleted because records were merged for the
evaluation of column functions with GROUP BY.

Field Name: QW0096RL

PASSES

The number of merge passes during sort processing.

Field Name: QW0096MP

SORT KEYS

The number of sort keys.

Field Name: QW0096SK

STMTNO

The statement number for the query that invokes sort.

Field Name: QW0096SN

WORKFILES

The number of work files, equal to the degree of parallelism, that sort has
partitioned.

Field Name: QW0096PW

SORT COLUMNS

The number of sort columns.

Field Name: QW0096SC

PARTITIONING BY SORT

Indicates whether the sorted records were partitioned.

Field Name: QW0096PP

SORT IN ADDITION

Indicates whether the input records were only partitioned or partitioned and sorted:

YES   The records were only partitioned.
The records were partitioned and sorted.

Field Name: QW0096PO

TYPE

The type of sort that occurred. The possible values are:

- **ESA**: ORDER BY sort using the ESA sort hardware instructions
- **ESAG**: GROUP BY sort using the ESA sort hardware instructions
- **ESAT**: ESA tag sort using the ESA sort hardware instructions
- **RCYC**: GROUP RECYCLING sort using the ESA sort hardware instructions
- **REG**: Regular sort
- **NONE**: No sort occurred.

Field Name: QW0096TS

PARTITIONING OCCURRED

Indicates when partitioning took place:

- **W**: The work file was partitioned at the end of the input phase. No merge occurred.
- **M**: The output was partitioned during the last merge pass.
- **O**: One record was put into one partition.
- **P**: The records were presorted before being partitioned.
- **N**: The work file was not partitioned.

Field Name: QW0096PT
IFCID 097 - AMS Command End

This topic shows detailed information about “Record Trace - IFCID 097 - AMS Command End”.

Record trace - IFCID 097 - AMS Command End

The field labels shown in the following sample layout of “Record Trace - IFCID 097 - AMS Command End” are described in the following section.

RETURN 0 COMMAND: DEFINE DSNC210.DSNDBC.DB2PMDB1.DB2PMIX1.IO001.A001 CLUSTER CATALOG(DSNC210);

RETURN

The AMS return code.

Field Name: QW0097RC

COMMAND

The AMS command text.

Field Name: QW0097P1
This topic shows detailed information about “Record Trace - IFCID 098 - IBM Service Record”.

This record is for IBM service use.
This topic shows detailed information about “Record Trace - IFCID 099 - IBM Service Record”.

This record is for IBM service use.
IFCID 100 - IBM Service Record

This topic shows detailed information about “Record Trace - IFCID 100 - IBM Service Record”.

This record is for IBM service use.
IFCID 101 - IBM Service Record

This topic shows detailed information about “Record Trace - IFCID 101 - IBM Service Record”.

This record is for IBM service use.
This topic shows detailed information about “Record Trace - IFCID 102 - IBM Service Record”.

This record is for IBM service use.
IFCID 103 - SOS Off

This topic shows detailed information about “Record Trace - IFCID 103 - SOS Off”.

Record trace - IFCID 103 - SOS Off

The field labels shown in the following sample layout of “Record Trace - IFCID 103 - SOS Off” are described in the following section.

TIME: 3/18/92 14:25:37.400234

TIME

Store clock time.

Field Name: QW0103SC
IFCID 104 - Log Data Set

This topic shows detailed information about “Record Trace - IFCID 104 - Log Data Set”.

Record trace - IFCID 104 - Log Data Set

The field labels shown in the following sample layout of “Record Trace - IFCID 104 - Log Data Set” are described in the following section.

```
DSID ACTLG103
DSNAME DSNC310.LOGCOPY1.DS03
DSID
    The data set identifier of the active log manager.
    Field Name: QW0104DI

DSNAME
    The data set name of the active log.
    Field Name: QW0104DN
```
IFCID 105 - DBID/OBID Translation

This topic shows detailed information about “Record Trace - IFCID 105 - DBID/OBID Translation”.

This record contains up to 100 data sections. The following data is printed for each section in the record:

Record trace - IFCID 105 - DBID/OBID Translation

The field labels shown in the following sample layout of “Record Trace - IFCID 105 - DBID/OBID Translation” are described in the following section.

DBID: 5 DATABASE NAME: DSNDB07
OBID: 24 OBJECT NAME: DSNDSX02

DBID
The decimal identifier of the database.
Field Name: QW0105DB

DATABASE NAME
The database name.
Field Name: QW0105DN

OBID
The decimal identifier of the object. Examples of objects are table space and index space.
Field Name: QW0105OB

OBJECT NAME
The name of the object. Examples of objects are table space and index space.
Field Name: QW0105TN
IFCID 106 shows the data from system parameters.

"IFCID 106 - Application Programming Defaults" on page 40-361
This topic shows detailed information about "Record Trace - IFCID 106 - Application Programming Defaults".

"IFCID 106 - Data Sharing Parameters" on page 40-369
This topic shows detailed information about "Record Trace - IFCID 106 - Data Sharing Parameters".

"IFCID 106 - Databases/Spaces Automatically Deferred” on page 40-372
This topic shows detailed information about "Record Trace - IFCID 106 - Databases/Spaces Automatically Deferred”.

"IFCID 106 - Databases/Spaces Automatically Restarted” on page 40-373
This topic shows detailed information about "Record Trace - IFCID 106 - Databases/Spaces Automatically Restarted”.

"IFCID 106 - Databases/Spaces Automatically Started” on page 40-374
This topic shows detailed information about "Record Trace - IFCID 106 - Databases/Spaces Automatically Started”.

"IFCID 106 - Distributed Data Facility Parameters” on page 40-375
This topic shows detailed information about "Record Trace - IFCID 106 - Distributed Data Facility Parameters”.

"IFCID 106 - IRLM Processing Parameters” on page 40-379
This topic shows detailed information about "Record Trace - IFCID 106 - IRLM Processing Parameters”.

"IFCID 106 - Log Initialization Parameters (Part 1)” on page 40-382
This topic shows detailed information about "Record Trace - IFCID 106 - Log Initialization Parameters (Part 1)”.

"IFCID 106 - Log Initialization Parameters (Part 2)” on page 40-384
This topic shows detailed information about "Record Trace - IFCID 106 - Log Initialization Parameters (Part 2)”.

"IFCID 106 - Miscellaneous Installation Parameters” on page 40-389
This topic shows detailed information about "Record Trace - IFCID 106 - Miscellaneous Installation Parameters”.

"IFCID 106 - Stored Procedures Parameters” on page 40-423
This topic shows detailed information about "Record Trace - IFCID 106 - Stored Procedures Parameters”.

"IFCID 106 - System Initialization Parameters” on page 40-424
This topic shows detailed information about "Record Trace - IFCID 106 - System Initialization Parameters”.

"IFCID 106 - VSAM Catalog Name Qualifier” on page 40-441
This topic shows detailed information about "Record Trace - IFCID 106 - VSAM Catalog Name Qualifier”.
IFCID 106 - Application Programming Defaults

This topic shows detailed information about "Record Trace - IFCID 106 - Application Programming Defaults".

This block shows application programming defaults.

The values shown are used as default values by the program preparation panels, program preparation CLIST (DSNH), and precompiler. They can also be used as defaults by other programs, such as Query Management Facility (QMF).

Changing some of these defaults is not recommended because changes can make the syntax of existing SQL statements invalid or affect the way application programs run.

Values set here are contained in load module DSNHDECP, in library prefix.SDSNEXIT, which can be loaded and accessed by application programs. When modifying DSNHDECP, do so only by changing and running the installation CLIST.

Do not modify the data in DSNHDECP. If you modify any installation parameters by changing job DSNTIJUZ directly, these values are not recorded for later updates, new installations, or migrations.

Record trace - IFCID 106 - Application Programming Defaults

The field labels shown in the following sample layout of "Record Trace - IFCID 106 - Application Programming Defaults" are described in the following section.

<table>
<thead>
<tr>
<th>Field Label</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPLICATION PROGRAMMING DEFAULTS</td>
<td></td>
</tr>
<tr>
<td>VERSION</td>
<td>1110</td>
</tr>
<tr>
<td>DEFAULT SUBSYSTEM</td>
<td>DB1</td>
</tr>
<tr>
<td>DECFLOAT RND MODE</td>
<td>HALF EVEN</td>
</tr>
<tr>
<td>LOCAL DATE LENGTH</td>
<td>0</td>
</tr>
<tr>
<td>LOCAL TIME LENGTH</td>
<td>0</td>
</tr>
<tr>
<td>INSTALL TYPE</td>
<td>CURRENT</td>
</tr>
<tr>
<td>IMP TIMEZONE</td>
<td>CURRENT</td>
</tr>
<tr>
<td>QWPBLVL</td>
<td>VI1RM0</td>
</tr>
<tr>
<td>QWPBCHAR</td>
<td>ALPHANUM</td>
</tr>
<tr>
<td>QWPBREL</td>
<td>UNICOD, EBCDIC</td>
</tr>
<tr>
<td>EBCDIC SBCS CCSID</td>
<td>EBCDIC SBCS CCSID: 1148</td>
</tr>
<tr>
<td>DEFAULT HOST LANG</td>
<td>IBMCOB</td>
</tr>
<tr>
<td>DECIMAL POINT OPT</td>
<td>PERIOD</td>
</tr>
<tr>
<td>DEFLT DEC ARIHM</td>
<td>ASCII SBCS CCSID: 65534</td>
</tr>
<tr>
<td>DEFLT ENC SCHEME</td>
<td>EBCDIC SBCS CCSID: 65534</td>
</tr>
<tr>
<td>DEFLT MIXED GRAPH</td>
<td>NO ASCII SBCS CCSID: 65534</td>
</tr>
<tr>
<td>DEFLT SQL DELIM</td>
<td>APOST</td>
</tr>
<tr>
<td>DEFLT SQL DELIM</td>
<td>APOST</td>
</tr>
<tr>
<td>SQL LANG SUPP LVL</td>
<td>NO UNICOD SBCS CCSID: 367</td>
</tr>
<tr>
<td>USE FOR DYN RULES</td>
<td>YES UNICOD SBCS CCSID: 1200</td>
</tr>
<tr>
<td>IMP TIMEZONE(Hex)</td>
<td>X'9999999C'</td>
</tr>
<tr>
<td>IMP TIMEZONE</td>
<td>'BLANK'</td>
</tr>
<tr>
<td>IMP TIMEZONE</td>
<td>'BLANK'</td>
</tr>
<tr>
<td>IMP TIMEZONE</td>
<td>CURRENT</td>
</tr>
<tr>
<td>IMP TIMEZONE</td>
<td>CURRENT</td>
</tr>
</tbody>
</table>

VERSION

The version, release, and modification level.

Field Name: QWPBREL

DEFAULT SUBSYSTEM

The MVS subsystem name for DB2. The name is used in member IEFSSN xx of SYS1.PARMLIB.

A valid name has 1-4 characters, the first must be A-Z, #, $, or @. Others must be A-Z, 1-9, #, $, or @. Default is DSN1.

Install parameter SUBSYSTEM NAME on panel DSNTIPM, or ZPARM SSID in DSNHDECP.

Field Name: QWPBSSID

EBCDIC SBCS CCSID

The EBCDIC single-byte coded character set ID.
IFCID 106 - Application Programming Defaults

A coded character set identifier (CCSID) must be specified when DDF STARTUP OPTION field on panel DSNTIPR is set to AUTO or COMMAND, or when the MIXED DATA field on panel DSNTIPF is set to YES. When mixed data is used, valid Mixed Data CCSID must also be specified.

A nonexistent CCSID causes an error.

An incorrect CCSID can corrupt data.

Install parameter EBCDIC CCSID on panel DSNTIPF, or ZPARM SCCSID in DSNHDECP.

Field Name: QWPBSID

DEFAULT HOST LANGUAGE
The default programming language for your site. This can be:

• ASM
• C
• CPP
• COBOL
• COB2
• IBMCOB
• FORTRAN
• PLI

When this is C or C++, you can fold SQL identifiers to uppercase.

Install parameter LANGUAGE DEFAULT on panel DSNTIPF, or ZPARM DEFLANG in DSNHDECP.

Field Name: QWPBLANG

DECIMAL POINT OPT
Indicates whether the decimal contains a comma (,) or a period (.). This parameter is used for dynamic SQL and COBOL programs. It is not used or supported by other languages.

Install parameter DECIMAL POINT IS on panel DSNTIPF, or ZPARM DECIMAL in DSNHDECP.

Derivation: DB2 field QWPBDE

Field Name: QWPBDE

EBCDIC MBCS CCSID
The EBCDIC mixed coded character set ID.

A coded character set identifier (CCSID) must be specified when DDF STARTUP OPTION field on panel DSNTIPR is set to AUTO or COMMAND, or when the MIXED DATA field on panel DSNTIPF is set to YES. When mixed data is used, valid Mixed Data CCSID must also be specified.

A nonexistent CCSID causes an error.

An incorrect CCSID can corrupt data.

Install parameter EBCDIC CCSID on panel DSNTIPF, or ZPARM MCCSID in DSNHDECP.
Field Name: QWPBMID

DECFLOAT RND MODE

The default rounding mode for the decimal floating point type. Possible values are:

X'80'  ROUND_CEILING
X'40'  ROUND_DOWN
X'20'  ROUND_FLOOR
X'10'  ROUND_HALF_DOWN
X'08'  ROUND_HALF_EVEN
X'04'  ROUND_HALF_UP
X'02'  ROUND_UP

Otherwise this field shows 'BLANK'.

ZPARM DEF_DECFLOAT_ROUND_MODE in DSNHDECP.

Field Name: QWPBDDRM

DEFLT ENC SCHEME

The default encoding scheme, which can be ASCII or EBCDIC, or UNICODE.

Install parameter DEF ENCODING SCHEME on panel DSNTIPF, or ZPARM ENSCHEME in DSNHDECP.

Derivation: DB2 field QWPBENS

Field Name: QWPBENS

EBCDIC GBCS CCSID

The EBCDIC graphic coded character set ID.

A coded character set identifier (CCSID) must be specified when DDF STARTUP OPTION field on panel DSNTIPR is set to AUTO or COMMAND, or when the MIXED DATA field on panel DSNTIPF is set to YES. When mixed data is used, valid Mixed Data CCSID must also be specified.

A nonexistent CCSID causes an error.

An incorrect CCSID can corrupt data.

Install parameter EBCDIC CCSID on panel DSNTIPF, or ZPARM GCCSID in DSNHDECP.

Field Name: QWPBGID

DEFAULT DELIMITER

Shows the string delimiter for COBOL. Default string delimiter is the quotation mark. This option is applicable to all types of COBOL.

Install parameter STRING DELIMITER on panel DSNTIPF, or ZPARM DELIM in DSNHDECP.

Field Name: QWPBDL

DISTR SQL STR DEL
The SQL delimiter for bind operations at this DB2 if the requester does not provide DB2 with this information.

Field Name: QWPBDSD

ASCII SBCS CCSID

The ASCII single-byte coded character set ID.
The default (0) means the installation has no ASCII databases, table spaces, or tables.
Install parameter ASCII CCSID on panel DSNTIPF, or ZPARM ASCCSID in DSNHDECP.

Field Name: QWPBASID

DEFLT SQL DELIMIT

The string delimiter for SQL.
Install parameter SQL STRING DELIMITER on panel DSNTIPF, or ZPARM SQLDELI in DSNHDECP.

Derivation: DB2 field QWPBSDL

Field Name: QWPBSDL

DEFLT DEC ARITHM

Indicates the rules of precision for a decimal field.
Install parameter DECIMAL ARITHMETIC on panel DSNTIP4, or ZPARM DECARTH in DSNHDECP.

Derivation: DB2 field QWPBAR

Field Name: QWPBAR

ASCII MBCS CCSID

Indicates the ASCII mixed coded character set ID.
The default (0) means the installation has no ASCII databases, table spaces, or tables.
Install parameter ASCII CCSID on panel DSNTIPF, or ZPARM AMCCSID in DSNHDECP.

Field Name: QWPBAMID

LOCAL DATE LENGTH

Shows the length of the longest field required to hold a locally defined date.
The default (0) indicates an IBM-supplied format (ISO, JIS, USA, or EUR).
Install parameter LOCAL DATE LENGTH on panel DSNTIP4, or ZPARM DATELEN in DSNHDECP.

Field Name: QWPBDLEN

DEFLT MIXED GRAPH

Indicates whether the code points X'0E' and X'0F' are the shift-out and shift-in controls for character strings that include double-byte characters.
Install parameter MIXED DATA on panel DSNTIPF, or ZPARM MIXED in DSNHDECP.
IFCID 106 - Application Programming Defaults

Field Name: QWPBGRA

ASCII GBCS CCSID
Indicates the ASCII graphic coded character set ID.
The default (0) means the installation has no ASCII databases, table spaces, or tables.
Install parameter ASCII CCSID on panel DSNTIPF, or ZPARM AGCCSID in DSNHDECP.

Field Name: QWPBAGID

LOCAL TIME LENGTH
Shows the length of the longest field required to hold a time when a locally defined time format is used.
The default (0) indicates an IBM-supplied format (ISO, JIS, USA, or EUR).
Install parameter LOCAL TIME LENGTH on panel DSNTIP4, or ZPARM TIMELEN in DSNHDECP.

Field Name: QWPBTLEN

SQL LANG SUPP LVL
Shows whether SQL, the language standard used by applications, conforms to 1986 ANSI SQL standard.
YES Conforms to the 1986 ANSI SQL standard
NO Conforms to the SQL language defined by DB2
86 Conforms to the 1986 ANSI SQL standard
Install parameter STD SQL LANGUAGE on panel DSNTIP4, or ZPARM STDSQL in DSNHDECP.

Field Name: QWPBSQL

UNICODE SBCS CCSID
Unicode Single Byte Character Set identification.
Parameter UNICODE CCSID in installation panel DSNTIPF, or ZPARM USCCSID in macro DSNHDECP.

Field Name: QWPBUSID

DATE FORMAT
Default output format for dates.
Valid formats are ISO (yyyy-mm-dd), USA (mm/dd/yyyy), EUR (dd.mm.yyyy), JIS (yyyy-mm-dd), or LOCAL (your choice, defined by a date exit routine). DB2 interprets the input date from the punctuation and converts the output date to the required format.
Install parameter DATE FORMAT on panel DSNTIP4, or ZPARM DATE in DSNHDECP.

Field Name: QWPBDATE

USE FOR DYN RULES
ICFID 106 - Application Programming Defaults

Shows whether DB2 uses the application programming defaults specified on this panel or those of the DB2 precompiler options for dynamic SQL statements bound using DYNAMICRULES bind, define, or invoke behavior.

When YES, the application programming (DSNHDECP) defaults are used for dynamic SQL statements in plans or packages bound using DYNAMICRULES bind, define, or invoke behavior.

The following defaults are affected:

- DECIMAL POINT IS
- STRING DELIMITER
- SQL STRING DELIMITER
- MIXED DATA
- DECIMAL ARITHMETIC

When NO, values of the precompiler options are used for dynamic SQL statements in plans or packages bound with DYNAMICRULES(BIND).

Install parameter USE FOR DYNAMICRULES on panel DSNTIP4, or ZPARM DYNRULS in DSNHDECP.

**Field Name:** QWPBDRLS

UNICODE MBCS CCSID

Unicode Mixed Character Set identification.

Parameter UNICODE CCSID in installation panel DSNTIPF, or ZPARM UMCCSID in macro DSNHDECP.

**Field Name:** QWPBUMID

TIME FORMAT

Indicates the default output format for times.

Valid values are ISO (hh:mm:ss), USA (hh:mm AM), EUR (hh:mm.ss), JIS (hh:mm:ss), or LOCAL (your choice, defined by a time exit routine). DB2 interprets the input time from the punctuation and converts the output time to the required format.

Install parameter TIME FORMAT on panel DSNTIP4, or ZPARM TIME in DSNHDECP.

**Field Name:** QWPBTIME

APPLIC ENCODING

Application encoding scheme.

Install parameter APPLICATION ENCODING on installation panel DSNTIPF, or ZPARM APPENSCH in DSNHDECP.

**Field Name:** QWPBAPSC

UNICOD GBCS CCSID

Unicode graphics character set identification.

Parameter UNICODE CCSID in installation panel DSNTIPF, or ZPARM UGCCSID in macro DSNHDECP.

**Field Name:** QWPBUGID

INSTALL TYPE
IFCID 106 - Application Programming Defaults

If YES, the DB2 subsystem/group is running in New Function Mode. At this mode/catalog level, the New Function Mode is enabled and available. The DB2 catalog is completely Unicode (UTF-8) and long names can be used.

Install parameter INSTALL TYPE on panel DSNTIPA1, or ZPARM NEWFUN in DSNHDECP.

**Field Name:** QWPBNEWF

**PAD NULL-TERMIN**

Shows whether output host variables that are NULL-terminated strings are padded with blanks and a NULL terminator.

When NO, NULL-terminated output host variables have the NULL terminator placed at the end of actual data returned in the host variable. When YES, NULL-terminated output host variables have the NULL terminator placed at the end of the string, after the string has been padded with blanks from the end of the actual data to the declared length of the output host variable.

Install parameter PAD NUL-TERMINATED on installation panel DSNTIP4, or ZPARM PADNTSTR in DSNHDECP.

**Field Name:** QWPBPAD

**DB2 DECP INDICAT**

Indicates that DECP is supplied by DB2.

Using a DB2 supplied DECP could cause data corruption due to applications using wrong CCSIDs.

**Field Name:** QWPDB2S

**IMPL TIMEZONE (HEX)**

The implicit time zone that is associated with DB2 table columns and routine parameters that are declared as time stamp with time zone.

For IFCID 106 - Application Programming Defaults, this field is displayed twice, with its hex value and in a readable string.

This field corresponds to DSNHDECP field IMPLICIT_TIMEZONE.

**Field Name:** QWPBIMTZ

**DEFAULT LOCALE**

The system LOCALE LC_CTYPE.

A locale is the part of the system environment that depends on language and cultural conventions. An LC_TYPE is a subset of a locale that applies to character functions. The UPPER, LOWER, and TRANSLATE scalar functions use the CURRENT LOCALE LC_CTYPE system default or special register. The results of these functions can vary, depending on the setting of the locale.

The following values are possible:

**BLANK**

The source field is empty.
IFCID 106 - Application Programming Defaults

This is the default, unless it is necessary to run the UPPER, LOWER, or TRANSLATE functions for data that must be interpreted using the rules provided by specific locales, for example, En_US or Fr_CA.

1st word
The source field contains left-justified word(s), where each byte of a word is > X'40'. It can be a single word or several ones, delimited by bytes <= X'40'..

Note: These hexadecimal codes do not represent printable characters.

N/P The source field contains regular words that are not left-justified. This means that the first bytes are <= X'40'. N/P is also shown if the whole source field only consists of bytes < X'40', such as zeros.

Install parameter LOCALE LC_CTYPE on panel DSNTIPF, or ZPARM LC_TYPE in DSNHDECP.

Field Name: QWPBLCTP

IMPL TIMEZONE
The implicit time zone that is associated with DB2 table columns and routine parameters that are declared as time stamp with time zone.

For IFCID 106 - Application Programming Defaults, this field is displayed twice, with its hex value and in a readable string.

This field corresponds to DSNHDECP field IMPLICIT_TIMEZONE.

Field Name: QWPBLMTZ

QWPBLVL
This field is for IBM service.

Field Name: QWPBLVL

QWPBCHAR
Shows the default character set, ALPHANUM or KATAKANA.
ZPARM CHARSET in DSNHDECP.

Field Name: QWPBCHAR
IFCID 106 - Data Sharing Parameters

This topic shows detailed information about “Record Trace - IFCID 106 - Data Sharing Parameters”.

This block shows the members in a data-sharing group.

DB2 subsystems that share data must belong to a DB2 data sharing group, which runs on a Parallel Sysplex®. A data sharing group is a collection of one or more DB2 subsystems that access shared DB2 data. A Parallel Sysplex is a collection of MVS systems that communicate and cooperate with each other.

Record trace - IFCID 106 - Data Sharing Parameters

The field labels shown in the following sample layout of “Record Trace - IFCID 106 - Data Sharing Parameters” are described in the following section.

GROUP NAME

The name of the DB2 data sharing group.

The group name encompasses the entire data sharing group and is the basis for the coupling facility structure names.

N/A means this DB2 is not part of a data sharing group.

Install parameter GROUP NAME on panel DSNTIPK, or ZPARM GRPNAME in DSN6GRP.

Field Name: QWPAGRPN

MEMBER NAME

The member name of this DB2.

N/A means this DB2 is not part of a data sharing group.

Install parameter MEMBER NAME on panel DSNTIPK, or ZPARM MEMBNAME in DSN6GRP.

Field Name: QWPAMBRN

DATA SHARING ENAB

Indicates whether data sharing is enabled.

Field Name: QWPADSHR

MAX # OF MEMBERS

The maximum number of members possible in a data sharing group. This is a constant (248) and is not shown on any installation panel.

Field Name: QWPAMAXM

IMMEDWRITE FLAG

Indicates how DB2 updates group buffer pool dependent pages. This is only valid in a data-sharing environment.

Group buffer pool dependent pages can be written to DASD or SYSTEM pagesets.
IFCID 106 - Data Sharing Parameters

Values shown are:

**NO**  DB2 uses normal write activity for updates, this is the default. Pages are written out at, or before phase 2 commit, or at the end of an abend for transactions that have rolled back.

**PH1**  Pages are written out at, or before phase 1 commit.

If a transaction subsequently rolls back, the pages are updated in the group buffer pool at the end of the rollback and are written out at the end of the abend.

**YES**  Pages are written out to the coupling facility as soon as the buffer update commits. Pages are written out regardless of whether the update occurs during forward progress or rollback of the transaction.

This option can affect performance due to coupling facility overhead.

Install parameter IMMEDIATE WRITE on panel DSNTIP8, or ZPARM IMMEDWRI in DSN6GRP.

**Field Name:** QWPAIMMW

**CONVERSION FACTOR**

The CPU service unit conversion factor for this CPU.

This factor allows conversion CPU time in seconds to a common unit, called service unit (SU). The conversion factor used depends on the machine. Service units allow you to calculate CPU execution times across a data sharing group.

The conversion factor is used as follows:

\[
\text{CP secs} \times 16,000,000 \div \text{Conversion Factor} = \text{SUs}
\]

\[
\text{SUs} \times \text{Conversion Factor} \div 16,000,000 = \text{CP secs}
\]

This field does not map to an installation panel.

**Field Name:** QWPASUCV

**QWPACOOR**

Shows whether this DB2 member can coordinate parallel processing on other members of the group.

When NO, a query can be processed by this DB2 member only.

When YES, a read-only query running on this DB2 member can be processed in part on other members of the group.

N/A means this DB2 is not part of a data sharing group.

Install parameter COORDINATOR on panel DSNTIPK or ZPARM COORDNTR in DSN6GRP.

**Field Name:** QWPACOOR

**QWPAAST**

Shows whether this DB2 member can assist a parallelism coordinator with parallel processing.

When YES, this member is considered an assistant at both bind and run time. To be a viable assistant at run time, both the VPPSEQT and VPXPSEQT buffer pool thresholds of this member must be greater than 0.
IFCID 106 - Data Sharing Parameters

N/A means this DB2 is not part of a data sharing group.
Install parameter ASSISTANT on panel DSNTIPK or ZPARM ASSIST in DSN6GRP.

Field Name: QWPAASST
IFCID 106 - Databases/Spaces Automatically Deferred

IFCID 106 - Databases/Spaces Automatically Deferred
This topic shows detailed information about “Record Trace - IFCID 106 - Databases/Spaces Automatically Deferred”.

Record trace - IFCID 106 - Databases/Spaces Automatically Deferred

The field labels shown in the following sample layout of “Record Trace - IFCID 106 - Databases/Spaces Automatically Deferred” are described in the following section.

DATABASES/SPACES STARTED AUTOMATICALLY ALL

Tablespace Names

Contains the name of a table space or index space that is to be started automatically.

Field Name: QWP8SPNM
IFCID 106 - Databases/Spaces Automatically Restarted

This topic shows detailed information about “Record Trace - IFCID 106 - Databases/Spaces Automatically Restarted”.

Database Names

The name of a database that is to be started automatically.

Field Name: QWP8DBNM
IFCID 106 - Databases/Spaces Automatically Started

Record trace - IFCID 106 - Databases/Spaces Automatically Started

The field labels shown in the following sample layout of “Record Trace - IFCID 106 - Databases/Spaces Automatically Started” are described in the following section.

DATABASES/SPACES STARTED AUTOMATICALLY
ALL

Database names
The name of a database that is to be started automatically.
Field Name: QWP8DBNM

Tablespace names
Contains the name of a table space or index space that is to be started automatically.
Field Name: QWP8SPNM
IFCID 106 - Distributed Data Facility Parameters

This topic shows detailed information about “Record Trace - IFCID 106 - Distributed Data Facility Parameters”.

This block shows how Distributed Data Facility (DDF) was started and the protocols used.

To use DDF, you must have VTAM installed, even if you use TCP/IP connections only.

Record trace - IFCID 106 - Distributed Data Facility Parameters

The field labels shown in the following sample layout of “Record Trace - IFCID 106 - Distributed Data Facility Parameters” are described in the following section.

FACILITY NAME: The name of the DDF facility.

Field Name: QWP9NAME

RLF ERROR ACTION

Shows what DB2 does when the governor cannot access the resource limit specification table or when no row in the table matches the currently executing statement.

NOLIMIT (default) allows all dynamic SQL statements to run without limit.

NORUN terminates all dynamic SQL statements immediately with an SQL error code.

A number from 1 to 5000000 is the default limit; if the limit is exceeded, the SQL statement is terminated.

Install parameter RLST ACCESS ERROR on panel DSNTIPR, or ZPARM RLFERRD in DSN6FAC.

Field Name: QWP9RLER

RESYNCH INTERVAL

The number of minutes between resynchronization periods.

A resynchronization period is the time during which indoubt logical units of work involving this DB2 subsystem and partner logical units are processed.

Install parameter RESYNC INTERVAL on panel DSNTIPR, or ZPARM RESYNC in DSN6FAC.

Field Name: QWP9RYC

TCP/IP VERIFIED

Indicates whether DB2 accepts TCP/IP connection requests containing only a user ID.
IFCID 106 - Distributed Data Facility Parameters

When YES, a connection request is accepted with a user ID only. This value must be the same for all members of a data sharing group.

When NO (default), TCP/IP clients must provide authentication information (password, RACF PassTicket, or Kerberos ticket) to gain access to DB2.

Install parameter TCP/IP ALREADY VERIFIED on panel DSNTIP5, or ZPARM TCPALVER in DSN6FAC.

**Field Name:** QWP9TCPA

**FACILITY START**

Indicates whether DDF is loaded, and if so, how it was started.

When NO, DDF was not loaded at DB2 startup and cannot be started.

AUTO means DDF was loaded and started automatically when DB2 was started. The DDF address space was started as part of DDF initialization.

COMMAND means DDF was initialized and the DDF address space was started at DB2 startup. If DDF is running, it was started from the console with the -DSN1 START DDF command. If it is not running, it can be started with this command.

Install parameter DDF STARTUP OPTION on panel DSNTIPR, or ZPARM DDF in DSN6FAC.

**Field Name:** QWP9STRT

**IDLE THR TIMEOUT**

The approximate time, in seconds, that an active server thread can remain idle before it is canceled.

Inactive and indoubt threads are not subject to timeout.

Threads are checked for timeouts every 3 minutes. This means that timeouts might not be honored for up to 3 minutes when the timeout value is less than this.

0 (default) means timeout processing is disabled, idle server threads remain in the system and continue to hold their resources, if any.

Install parameter IDLE THREAD TIMEOUT on panel DSNTIPR, or ZPARM IDTHTOIN in DSN6FAC.

**Field Name:** QWP9TTO

**DBAT STATUS**

Shows whether DB2 inactivates threads that have successfully committed or rolled back, and hold no cursors.

ACTIVE provides the best performance but consumes system resources.

INACTIVE is recommended when the installation must support a large number of connections.

When a thread becomes eligible for inactivation, DB2 tries to make it a type 2 inactive thread, which uses less storage than a type 1 inactive thread. If this fails, DB2 tries to make it a type 1 inactive thread. If neither attempt is successful, the thread remains active.

Install parameter DDF THREADS on panel DSNTIPR, or ZPARM CMTSTAT in DSN6FAC.
IFCID 106 - Distributed Data Facility Parameters

Field Name: QWP9CMST

TCP/IP KEEPALIVE

Indicates whether the TCP/IP configuration KeepAlive value has been overwritten.

When ENABLE (default), KeepAlive is enabled, the TCP/IP configuration stack value is used.

When DISABLE, TCP/IP KeepAlive has been disabled.

A value in the range 1 through 65534 means KeepAlive is active, and the TCP/IP stack value has been overridden. The number reported shows the time, in seconds, between TCP/IP probes.

When considering overwriting the keep-alive time, it is recommended to set a value close to the IDLE THREAD TIMEOUT value on installation panel DSNTIPR or the IRLM RESOURCE TIMEOUT value on installation panel DSNTIPI. It is good practice to set all these to about five minutes, or less.

Because KeepAlive detection is accomplished by probing the network at this interval, avoid small values, which can cause excessive network traffic and system resource consumption.

The trick is to find a proper balance that allows network failures to be detected on a timely basis without impacting system and network performance.

Install parameter TCP/IP KEEPALIVE on panel DSNTIP5, ZPARM TCPKPALV in DSN6FAC.

Field Name: QWP9TCKA

MAX T1 INACT THR

Indicates the number of type 1 inactive threads that DB2 allows.

A large number of type 1 inactive threads can adversely affect system performance. Type 1 inactive threads are used for DB2 private protocol.

DRDA uses type 2 inactive threads.

Zero indicates that type 1 inactive connections are not allowed. Threads remain active when they become eligible to be made a type 1 inactive thread.

A value greater than zero indicates that type 1 inactive connections are allowed, but are limited to this number. When a thread becomes eligible to be made a type 1 inactive thread, and this threshold is reached, the remote connection is terminated.

When this is equal to MAX REMOTE CONNECTED on panel DSNTIPE, DB2 allows all remote threads to become type 1 inactive threads.

Install parameter MAX INACTIVE DBATS on panel DSNTIPR, or ZPARM MAXTYPE1 in DSN6FAC.

Field Name: QWP9MAX1

POOL THR TIMEOUT

The approximate time, in seconds, that a DBAT can remain idle in the pool before it is terminated.
IFCID 106 - Distributed Data Facility Parameters

A DBAT thread in the pool counts as an active thread against MAX REMOTE ACTIVE and can hold locks, but does not have any cursors.

Threads are checked for timeouts every 3 minutes. This means that timeouts might not be honored for up to 3 minutes when the timeout value is less than this. The default is 120.

Install parameter POOL THREAD TIMEOUT on panel DSNTIP5, ZPARM POOLINAC in DSN6FAC.

Field Name: QWP9INAC

CONN Q MAX DEPTH

The maximum depth of the connection-request queue of connections that are waiting for a DBAT to process a request. If this value is non-zero, and QWP9CMST is active, or the subsystem is not a member of a data sharing group, DB2 operates as if this value were 0. This field corresponds to field CONN QUEUE MAX DEPTH on installation panel DSNTIP5. The ZPARM name is MACONQN in DSN6FAC.

Field Name: QWP9MCONQN

CONN Q MAX WAIT

The maximum time that a connection waits for a DBAT request. If this value is non-zero, and QWP9CMST is active, or the subsystem is not a member of a data sharing group, DB2 operates as if this value is 0.

This field corresponds to field CONN QUEUE MAX WAIT on installation panel DSNTIP5. The ZPARM name is MAXCONQW in DSN6FAC.

Field Name: QWP9MCONQW

DDF COMPATIBILITY

YES indicates that pre-DB2 10 behavior is used to determine the SQL types of stored procedure parameters in calls from non-Java clients. The ZPARM name is DDF_COMPATIBILITY and the ZPARM value is SP_PARM_NJv in DSN6FAC.

Field Name: QWP9SPPM
## PC SPECIFIED

Shows whether the IRLM uses the cross-address-space program call. This parameter determines where the IRLM lock control block structure is stored.

If you run a tightly-controlled environment and virtual storage is not constrained, use PC=NO. PC=YES is the conservative choice where insufficient information about the environment is available to make a well-informed decision.

With PC=NO, locks are managed in extended common service area (ECSA) and it is possible to achieve better CPU performance, because DB2 does not use cross-memory services for IRLM requests. However, ECSA is a limited resource and constrains the size of the private address space area available above the 16-MB line. The demand for ECSA storage to support locks may be excessive when one or more of the following conditions are true:

- Extensive use of row-level locking
- Ineffective lock avoidance
- Infrequent application commits
- Lock escalation via NUMLKTS and LOCKMAX is disabled because the applications cannot tolerate the impact
- Effectively no limit on the number of locks taken by an application (NUMLKUS is set very high)
- Multiple DB2 subsystems with IRLM PC=NO reside on the same z/OS image

Assuming the average lock consumes 536 bytes of storage, a single application which takes 100000 locks before a commit would consume almost 52 MB of ECSA when IRLM is configured with PC=NO. MAXCSA would have to be set to at least 52 MB. If a very large number of locks are held by concurrent application processes, the demand for ECSA may not be able to be supported.

Recommendation: If you run applications that have many of the above characteristics, it is strongly recommended to use PC=YES. Certain ERP vendor applications that run concurrent processes can acquire a very large number of held locks that would require a very large setting for MAXCSA, or cause an ECSA overflow which would adversely impact the availability of the z/OS image.

If PC=NO is selected, MAXCSA should be sized to support the concurrent number of held locks required and to avoid an ECSA overflow condition. When setting MAXCSA, check to ensure that the ECSA setting in
PARMLIB is sufficient to support the aggregate demand from IRLM and other subsystems. The ECSA size for z/OS is specified by the CSA keyword in the IEASSYSn member in SYS1.PARMLIB.

With PC=YES, locks are managed in the extended private area of the IRLM address space. This can increase the CPU cost of lock and unlock requests relative to PC=NO. However, with reasonable lock avoidance, the total CPU overhead is likely to be limited to 1 to 2%, which is well within measurement noise and therefore not significant.

With PC=YES, the MAXIMUM ECSA option is ignored but must not be zero. The amount of storage allowed for LOCK usage is determined from the extended storage provided to the IRLM address space at startup time. This amount is reduced by 200 MB to allow a buffer for IRLM and z/OS required storage and for DMBS MUST COMPLETE processes. The amount being monitored can be seen in the display message from the irlmproc,STATUS,STOR command. IRLM still uses CSA and ECSA for other purposes. If you need to create a dump for DB2 diagnostic purposes, you need to ensure that IRLM is included in the dump, and that the dump data sets are large enough to hold IRLM.

PC=NO is a good solution when one or more of the following conditions are true, particularly when running a data sharing configuration:
- Optimal CPU performance is required
- No constraint is necessary on available ECSA
- Significant IRLM lock contention and a very large number of lock requests with ineffective lock avoidance
- Relatively high IRLM SRB time

YES puts the lock control block structure in the IRLM private address space, and the program call instruction is used to address it. IRLM still uses CSA and ECSA for other purposes. With PC=YES, the MAXIMUM ECSA option is ignored.

Field Name: QWP5PCY
DEADLOCK WAIT
Wait time for local deadlock.
Field Name: QWP5DLOK
LOCAL/GLOBAL CYCL
Number of local cycles per global cycle.
Field Name: QWP5DCYC
TIMEOUT INTERVAL
Timeout interval.
Field Name: QWP5TVAL
MAX CSA USAGE
The maximum amount of common service area that can be used by IRLM.

The amount of space needed for the common service area (CSA) below the 16 MB line is less than 40 KB for each DB2 subsystem and 24 KB for each IRLM. High concurrent activity, parallelism, or high contention can require more CSA.
Most of the DB2 common data resides in the extended common service area (ECSA). Most modules, control blocks, and buffers reside in the extended private area. A DB2 subsystem with 200 concurrent users and 2000 open data sets should need less than 2 MB of virtual storage below the 16 MB line.

**Field Name:** QWP5MCSA

**LOCKTAB HASH ENTR**

The number of z/OS lock table hash entries.

**Field Name:** QWP5HASH

**MAX 31-BIT STOR**

The maximum amount of 31-bit IRLM private storage that is available of the 2 GB virtual storage limit, for normal operations in IRLM. IRLM reserves an additional 10% of the 2 GB for use by requests in IRLM.

**Field Name:** QWP5BPM

**PENDING HASH ENTR**

The number of z/OS lock table hash entries pending.

**Field Name:** QWP5PHSH

**LOCKTAB LIST ENTR**

The number of z/OS lock table list entries.

**Field Name:** QWP5RLE

**MAX 64-BIT STOR**

The maximum amount of 64-bit IRLM private storage that is available of the total amount of storage that is specified by MEMLIMIT, for normal operations in IRLM. IRLM reserves an additional 10% of the amount that is specified by MEMLIMIT for use by requests in IRLM.

**Field Name:** QWP5APM
IFCID 106 - Log Initialization Parameters (Part 1)

IFCID 106 - Log Initialization Parameters (Part 1)

This topic shows detailed information about “Record Trace - IFCID 106 - Log Initialization Parameters (Part 1)”.

Record trace - IFCID 106 - Log Initialization Parameters (Part 1)

The field labels shown in the following sample layout of “Record Trace - IFCID 106 - Log Initialization Parameters (Part 1)” are described in the following section.

LOG OUTPUT BUFFER

The output log buffer size in kilobytes.

There is only one output log buffer per DB2 subsystem.

Increasing this parameter reduces BSDS I/O updates when there is a buffer wraparound. Frequent wraparounds are likely in LOAD or REORG with logging, and mass insert operations.

Increasing this parameter also helps avoid log write waits for an available buffer during heavy update workload.

When the specified size is not a 4 KB multiple, it is rounded up to the next 4 KB multiple.

Install parameter OUTPUT BUFFER on DSNTIPL, or ZPARM OUTBUFF in DSN6LOGP.

Field Name: QWP2OBPS

MAX ARCHIVE INPUT UNITS

The maximum number of archive log volumes that can be allocated at the same time.

Field Name: QWP2INLM

INITIAL OPTIONS

The active log and BSDS initialization options in hexadecimal:

'80' Dual active logs
'40' Offload facility is on
'20' Dual BSDS are used

Field Name: QWP2OPT1

DEALLOC TIME(MIN)

The number of minutes an archive read tape unit can remain unused before it is deallocated.

When archive log data is read from tape, this value should be high enough to allow DB2 to optimize tape handling for multiple read applications.

Install parameter DEALLOC PERIOD on panel DSNTIPA, or ZPARM DEALLCT in DSN6LOGP.

Field Name: QWP2DMIN
READ COPY2 ARCHIVE
Indicates whether COPY2 archives should be read first when the DB2 subsystem is started. The default is NO. Install parameter READ COPY2 ARCHIVE on panel DSNTIPO, or ZPARM TRKRSITE in DSN6SPRM.

Field Name: QWP2ARC2

MAX ARCHIVE IN BSDS
The maximum number of archive log volumes that can be recorded in the BSDS.

When this number is exceeded, recording resumes at the beginning of the BSDS.

For dual archive, this value applies to each log data set. As an example, a value of 500 allows 500 COPY-1 and 500 COPY-2 data sets in the BSDS.

You must create image copies of all DB2 objects, probably several times, before the archive log data sets are discarded. If you fail to retain an adequate number of archive log data sets for all the image copies, you might need to cold start or reinstall DB2. In either case, data is lost.

Install parameter RECORDING MAX on panel DSNTIPA, or ZPARM MAXARCH in DSN6LOGP.

Field Name: QWP2ARCL

ARCHIVE OPTIONS
This field indicates whether the COPY2 archives should be read first when the DB2 subsystem is started.

Install parameter READ COPY2 ARCHIVE on DSNTIPO, or ZPARM ARC2FRST in DSN6LOGP.

Field Name: QWP2OPT2

DEALLOC TIME(SEC)
The deallocation time in seconds.

Field Name: QWP2DSEC
ICCID 106 - Log Initialization Parameters (Part 2)

This topic shows detailed information about “Record Trace - ICCID 106 - Log Initialization Parameters (Part 2)”.

Record trace - ICCID 106 - Log Initialization Parameters (Part 2)

The field labels shown in the following sample layout of “Record Trace - ICCID 106 - Log Initialization Parameters (Part 2)” are described in the following section.

LOG INITIALIZATION PARAMETERS (PART 2)

DATASET BLOCKSIZE: 24576
COPY1 DEVICE TYPE: DASD
MSS GROUP NAME 1: 'BLANK'
SECONDARY ALLOC.: 10
COPY2 DEVICE TYPE: 'BLANK'
MSS GROUP NAME 2: 'BLANK'
RETENTION PERIOD: 30
COPY1 PREFIX ....: DSNB11.ARCHLOG1
COPY2 PREFIX ....: DSNB11.ARCHLOG2
SINGLE VOLUME ....: NO
QUIESCE PERIOD ...: 5
CATALOG ARCH DS .: YES
SPACE ALLOC METHD: CYLINDER
ARCHLOG RACF PROT: NO
WTOR BEF ARCH MNT: YES
COMPACT DATA ....: NO
TS ARCHLOG DS ....: YES
QWP3LVL ........: 'X'C4E2DFIFIFIF040'
QWP3MLST ........:
0000 00000880 00000000 017E8000 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060
0020 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060
0040 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060
0060 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060
0080 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060 60606060

DATASET BLOCKSIZE

The block size of the archive log data set.

The block size must be compatible with the device type used for archive logs. The value is rounded up to the next multiple of 4096 bytes.

If the archive log is written to tape, use the largest possible block size to improve the reading speed.

Recommended block size values are 28672 for tape, 20480 for 3380, and 24576 for 3390 or RAMAC.

Install parameter BLOCK SIZE on panel DSNTIPA, or ZPARM BLKSIZE in DSN6ARVP.

Field Name: QWP3BKSZ

COPY1 DEVICE TYPE

The device type or unit name for storing archive log data sets.

The value can be any alphanumeric string. If you choose to archive to DASD, you can specify a generic device type with a limited volume range. DB2 requires that all archive log data sets allocated on DASD are cataloged.

If the device type is DASD, CATALOG DATA must be set to YES. If the unit name specifies DASD, the archive log data sets can extend to a maximum of 15 volumes. PRIQTY and SECQTY must be large enough to contain all active log data set data without extending beyond 15 volumes. If the unit name specifies a tape device, DB2 can extend to a maximum of 20 volumes. Default is TAPE.

Install parameter DEVICE TYPE 1 on panel DSNTIPA, or ZPARM UNIT in DSN6ARVP.

Field Name: QWP3UNT1

MSS GROUP NAME 1

The mass storage system volume group name of the first storage group.

Field Name: QWP3MSV1
PRIMARY ALLOCATION
The primary space allocation for archive data sets.
Install parameter PRIMARY QUANTITY on installation panel DSNTIPA, or ZPARM PRIQTY in DSN6ARVP.
Field Name: QWP3RISP

COPY2 DEVICE TYPE
Indicates the device type or unit name for storing the second copy of archive log data sets.
The value can be any alphanumeric string. If you choose to archive to DASD, you can specify a generic device type with a limited volume range. DB2 requires that all archive log data sets allocated on DASD are cataloged.
If the device type is DASD, then CATALOG DATA must be set to YES. If the unit name specifies DASD, the archive log data sets can extend to a maximum of 15 volumes. PRIQTY and SECQTY must be large enough to contain all active log data set data without extending beyond 15 volumes. If the unit name specifies a tape device, DB2 can extend to a maximum of 20 volumes. Default is TAPE.
Install parameter DEVICE TYPE 2 on panel DSNTIPA, or ZPARM UNIT2 in DSN6ARVP.
Field Name: QWP3UNT2

MSS GROUP NAME 2
The mass storage system volume group name of the second storage group.
Field Name: QWP3MSV2

SECONDARY ALLOC.
The amount of DASD secondary space allocation for an archive log data set.
The units used are specified by the ALLOCATION UNITS field. When blank (default), the CLIST calculates this space using block size and size of the log.
Install parameter SECONDARY QTY on panel DSNTIPA, or ZPARM SECQTY in DSN6ARVP.
Field Name: QWP3SECS

COPY 1 PREFIX
The prefix of the first archive log data set.
Install parameter Archive Logs: COPY1 PREFIX on panel DSNTIPH, or ZPARM ARCPFX1 in DSN6ARVP.
Field Name: QWP3RE1N

RETENTION PERIOD
The number of days DB2 keeps archive log data sets.
This value is added to the current date to calculate the expiration date.
IFCID 106 - Log Initialization Parameters (Part 2)

The retention period is often used in tape management systems to control the reuse and scratching of data sets and tapes. DB2 uses this as the value for the dynamic allocation parameter DALRETPD when archive log data sets are created.

Install parameter RETENTION PERIOD on panel DSNTIPA, or ZPARM ARCRETN in DSN6ARVP.

Field Name: QWP3RETN

COPY 2 PREFIX

The prefix of the second archive log data set. If single logging is used, this value is a default.

Install parameter Archive Logs: COPY2 PREFIX on panel DSNTIPH, or ZPARM ARCPFX2 in DSN6ARVP.

Field Name: QWP3RE2N

SINGLE VOLUME

Indicates whether single-volume DASD archives are used.

Install parameter SINGLE VOLUME on panel DSNTIPA, or ZPARM SVOLARC in DSN6ARVP.

Field Name: QWP3SVOL

QUIESCE PERIOD

The maximum amount of time (in seconds) permitted for DB2 to attempt a full system quiesce.

Install parameter QUIESCE PERIOD on panel DSNTIPA, or ZPARM QUIESCE in DSN6ARVP.

Field Name: QWP3MQP

CATALOG ARCH DS

The alias of the VSAM integrated catalog facility user catalog or the name of the master catalog where the DB2 VSAM data sets created during installation are cataloged. The MVS catalog alias is also used as the high-level qualifier for DB2 VSAM data sets.

Install parameter CATALOG ALIAS on panel DSNTIPA, or ZPARM CATALOG in DSN6ARVP.

Field Name: QWP3CTLG

SPACE ALLOC METHOD

The unit used in allocating archive data sets. Possible values are:

CYLINDER

Space allocation by cylinders (QWP3CYL=1)

TRACKS

Space allocation by tracks (QWP3TRCK=1)

BLOCKS

Space allocation by blocks (QWP3CYL=0 and QWP3TRCK=0)

Install parameter ALLOCATION UNITS on panel DSNTIPA, or ZPARM ALCUNIT in DSN6ARVP.

Field Name: RT0106SA
ARCHLOG RACF PROT

Indicates whether archive log data sets are protected with individual RACF profiles when they are created.

When YES, RACF protection must be active for DB2. YES also means that you cannot use RACF generic profiles for archive log data sets. If your archive log is on tape, RACF class TAPEVOL must be active, otherwise, the off-load will fail.

Install parameter ARCHIVE LOG RACF on panel DSNTIPP, or ZPARM PROTECT in DSN6ARVP.

Field Name: QWP3RTCT

WTOR BEF ARCH MNT

Indicates whether DB2 must send a message to the operator and wait for an answer before attempting to mount an archive log data set.

Other DB2 users can be forced to wait while the mount is pending. They are not affected while DB2 is waiting for a response to the message.

When YES, a device such as tape is used that requires long delays for mounts. DEVICE TYPE 1 shows the device type or unit name.

Install parameter WRITE TO OPER on panel DSNTIPA, or ZPARM ARCWTOR in DSN6ARVP.

Field Name: QWP3WTOR

COMPACT DATA

Indicates whether data written to archive logs is compacted.

This option only applies to data written to a 3480 device that has the improved data recording capability (IDRC) feature.

Install parameter COMPACT DATA on panel DSNTIPA, or ZPARM COMPACT in DSN6ARVP.

Field Name: QWP3COMP

TS ARCHLOG DS

Indicates whether the date and time of creation of the DB2 archive log data set is included in the archive log data set name.

Possible values are:

YES (QWP3DTIM=1)

The maximum allowable length of the user-controlled portion of the archive log prefix is reduced from 35 characters to 19 characters. This allows the 16-character timestamp to be added to the archive log data set prefix. The timestamp format is as follows: DyydddThhmmssst, where:

- D Starts the date.
- yy Is the last two digits of the year.
- ddd Is the day of the year.
- T Starts the time.
- hh Is the hour.
- mm Are the minutes.
Are the seconds.

Is the tenths of a second.

The maximum allowable length of the user-controlled portion of the archive log prefix is reduced from 35 characters to 19 characters. This reduction in size permits the 16-character date and time qualifiers (timestamp) to be added to the archive log data set prefix.

NO (QWP3DTIM=0 and QWP3DTFM=0)
The archive data set name does not contain a timestamp.

EXT (QWP3DTFM=1)
The archive data set name contains a timestamp with an extended date component in the format: .DYYYYYDDD. A value of EXT in this field causes the lengths of the values that are entered for field COPY 1 PREFIX and field COPY 2 PREFIX to be audited to ensure that neither exceeds 17 bytes (19 bytes for other settings of TIMESTAMP ARCHIVES).

Install parameter TIMESTAMP ARCHIVES on panel DSNTIPH, or ZPARM TSTAMP in DSN6ARVP.

Field Name: RT0106AL

QWP3LVL
This field is for IBM service.

Field Name: QWP3LVL

QWP3WLST
This field is for IBM service.

Field Name: QWP3WLST
IFCID 106 - Miscellaneous Installation Parameters

This topic shows detailed information about “Record Trace - IFCID 106 - Miscellaneous Installation Parameters”.

This block shows values that are not shown on DB2 installation panels. These values are either set internally by DB2, or calculated from other install parameter values.

When this block contains names that are too long for the space available, they are truncated. The full name is shown in the list of long names, which is printed at the end of this block. When present, the list shows the parameter identifier, in alphabetic order, and the complete name. If the name is too long for one line, it continues on the next line.

Record trace - IFCID 106 - Miscellaneous Installation Parameters

The field labels shown in the following sample layout of “Record Trace - IFCID 106 - Miscellaneous Installation Parameters” are described in the following section.
IFCID 106 - Miscellaneous Installation Parameters

EDM POOL SIZE

The size (in kilobytes) of the environmental descriptor manager (EDM) pool.

This can be the value calculated by the CLIST, based on input from previous panels, or the value entered in the Override column at installation time.

Install parameter EDMPOOL STORAGE SIZE on panel DSNTIPC, or ZPARM EDMPOOL in DSN6SPRM.

Field Name: QWP4EDPL

MVS ENVIRONMENT

The type of MVS environment in which DB2 is running.

Field Name: QWP4MVS

IRLM START TIME

The IRLM wait time in seconds.

DB2 autostart abends if IRLM does not start within this time.

Install parameter TIME TO AUTOSTART on panel DSNTIPI, or ZPARM IRLMSWT in DSN6SPRM.

Field Name: QWP4ISWT

DDL REGISTR FLAG

The DDL registration facility flag. It can have one of the following values:

X'80'  Data definition control support has been installed.
X'40'  The DB2 system is completely controlled by a set of closed applications.
IFCID 106 - Miscellaneous Installation Parameters

X'20'  Registered objects require fully qualified names.
X'00'  Reject the DDL that names an unregistered object.
X'1*'  Accept the DDL that names an unregistered object.
X'*1'  Reject the DDL that names an unregistered object if the current application is not registered.

Field Name: QWP4REGF

IRLM PROCEDURE

The name of the IRLM procedure invoked by MVS if AUTO START is YES.
The name cannot be the same as the subsystem name given for SUBSYSTEM NAME.
Install parameter PROC NAME on panel DSNTIPI, or ZPARM IRLMPRC in DSN6SPRM.

Field Name: QWP4IPRC

TAB OWNER

The owner of the application registration table and the object registration table.
This identifier can be a long string. If there is insufficient space to show the complete string, the string is truncated in the report block. The complete string is shown in a separate list of long names at the end of the report.
Install parameter REGISTRATION OWNER on panel DSNTIPZ, or ZPARM RGFCOLID in DSN6SPRM.

Field Name: QWP4REGC

IRLM MODULE NAME

The IRLM subsystem name defined to MVS.
This is used for communication between DB2 and the IRLM. It is included in the MVS subsystem table IEFSSN xx, where xx is the value of SUBSYSTEM MEMBER on installation panel DSNTIPM.
If the IRLM for IMS is installed, the DB2 IRLM name is different because two IRLMs on the same MVS system must have unique names.
Install parameter SUBSYSTEM NAME on panel DSNTIPI, or ZPARM IRLMSID in DSN6SPRM.

Field Name: QWP4ISID

APPL TABLE

The name of the application registration table.
Install parameter APPL REGISTRATION TABLE on panel DSNTIPZ or ZPARM RGFNMPRT in DSN6SPRM.

Field Name: QWP4REGA

MAXIMUM DATASETS

The maximum number of data sets that can be open at one time.
The practical limit can be less than the MVS limit of 32727, depending on available storage below the line.
IFCID 106 - Miscellaneous Installation Parameters

Install parameter DSMAX on panel DSNTIPC, or ZPARM DSMAX in DSN6SPRM.

Field Name: QWP4DSMX

INSTALL SYSADM

One of two authorization IDs with SYSADM authority. SYSADM users can access to DB2 in all cases.

This identifier can be a long string. If there is insufficient space to show the complete string, the string is truncated in the report block. The complete string is shown in a separate list of long names at the end of the report.

Install parameter SYSTEM ADMIN 1 on panel DSNTIPP, or ZPARM SYSADM in DSN6SPRM.

Field Name: QWP4SADM

IRLM INIT TIME

The number of seconds DB2 waits before querying whether IRLM has completed initialization.

DB2 parameter SPRMISWI in DSNMSPRM.

Field Name: QWP4ISWI

OBJ TABLE

The name of the object registration table.

Install parameter OBJT REGISTRATION TABLE on panel DSNTIPZ, or ZPARM RGFNMORT in DSN6SPRM.

Field Name: QWP4REGO

ASYNC DRAIN START

The percentage below 100% DSMAX that open data sets can reach before an asynchronous drain is started. The default is 1, meaning that asynchronous drain starts when the number of open data sets reaches 99% of DSMAX.

DB2 defers closing and deallocating the table spaces or indexes until the number of open data sets reaches one of the following limits:

• The MVS limit for the number of concurrently open data sets.
• 99% (default) of the value that you specified for DSMAX.

When one of these limits is reached, DB2 closes a number of data sets not in use equal to 3% (default) of the value DSMAX. Thus, DSMAX controls not only the limit of open data sets, but also the number of data sets that are closed when that limit is reached.

DB2 parameter SPRMTDD in DSN6SPRM.

Field Name: QWP4TDDN

DEFAULT USERID

The authorization ID used if RACF is not available for batch access and USER= is not specified in the job statement.

This identifier can be a long string. If there is insufficient space to show the complete string, the string is truncated in the report block. The complete string is shown in a separate list of long names at the end of the report.
IFCID 106 - Miscellaneous Installation Parameters

Install parameter UNKNOWN AUTHID on panel DSNTIPP, or ZPARM DEFLTID in DSN6SPRM.

Field Name: QWP4DFID

IRLM AUTOSTART

Indicates whether IRLM is started automatically by DB2.

Install parameter AUTO START on panel DSNTIPI, or ZPARM IRLMAUT in DSN6SPRM.

Field Name: QWP4IAUT

DATABASE NAME

The name of the database that contains the registration tables.

Install parameter REGISTRATION DATABASE on panel DSNTIPZ, or ZPARM RGFDBNAM in DSN6SPRM.

Field Name: QWP4REGN

ASYNC DRAIN STOP

The percentage of maximum open data sets until the asynchronous drain operations are stopped.

DB2 parameter SPRMMDD in DSN6SPRM.

Field Name: QWP4MDDN

SYSADM ID 2

One of two authorization IDs with SYSADM authority. SYSADM users can access to DB2 in all cases.

This identifier can be a long string. If there is insufficient space to show the complete string, the string is truncated in the report block. The complete string is shown in a separate list of long names at the end of the report.

Install parameter SYSTEM ADMIN 2 on panel DSNTIPP, or ZPARM SYSADM2 in DSN6SPRM.

Field Name: QWP4ADM2

IRLM TIMEOUT

The number of seconds before a timeout is detected.

This is an integer multiple of DEADLOCK TIME on panel DSNTIPJ.

Timeout means that a lock request has waited for a resource (or for claims on a resource for a particular claim class to be released) longer than this time.

For data sharing, the actual timeout period is longer than the timeout value.

Install parameter RESOURCE TIMEOUT on panel DSNTIPI, or ZPARM IRLMRWT in DSN6SPRM.

Field Name: QWP4TOUT

SITE TYPE

Shows whether this system is at a local site or a recovery site.
LOCALSITE
This is the site of the current system. Multiple image copies are made and are operational here. This is the default.

RECOVERYSITE
This an alternative site for recovery purposes.
The RECOVER utility uses this parameter to determine what site the current system is on and recovers everything from the copies of data registered at that site.
The RECOVER and MERGECOPY utilities use this to determine whether COPYDDN or RECOVERDDN is allowed with NEWCOPY NO.
Install parameter SITE TYPE on panel DSNTIPO, or ZPARM SITETYP in DSN6SPRM.
Field Name: QWP4MSTY

ENABLE DATA CAPT
Indicates whether change data capture is enabled.
Install parameter DPROP SUPPORT on panel DSNTIPO. ZPARM name is CHGDC in DSN6SPRM.
Field Name: QWP4CDC

SYSOPER ID
One of two authorization IDs with SYSOPR authority. SYSOPR users can access DB2 even if the DB2 catalog is unavailable.
This identifier can be a long string. If there is insufficient space to show the complete string, the string is truncated in the report block. The complete string is shown in a separate list of long names at the end of the report.
Install parameter SYSTEM OPERATOR 1 on panel DSNTIPP, or ZPARM SYSOPR1 in DSN6SPRM.
Field Name: QWP4OPR1

UTILITY FACTOR
Shows how much longer utilities can wait for a resource than SQL applications can.
This is the number of RESOURCE TIMEOUT units that a utility or utility command can wait for a lock or for all claims on a resource of a particular claim class to be released. The default value is 6, meaning a utility can wait 6 times longer than an SQL application for a resource.
Install parameter UTILITY TIMEOUT on panel DSNTIPI, or ZPARM UTIMOUT in DSN6SPRM.
Field Name: QWP4UTO

DDCS ESCAPE CHAR
The escape character used in the application registration table (ART) or object registration table (ORT).
Sets of names in the ART and ORT can be represented by patterns that use the underscore (_) and percent sign (%) characters in the same way as in an SQL LIKE predicate.
ENFORCE DPROP

Shows whether DataPropagator NonRelational (DPROP) is used to propagate SQL changes made to tables defined with DATA CAPTURE CHANGES.

1  No changes are propagated.

2  DPROP propagates SQL changes, and those changes made to tables defined with DATA CAPTURE CHANGES are only allowed when monitor trace class 6 is active, DPROP is installed, and the DB2 application is running in an IMS environment. If any of these conditions are not met, no changes to the DB2 table are permitted.

3  Data propagation occurs when monitor trace class 6 is active, DPROP is installed, and the DB2 application is running in an IMS environment. In this instance, an application that is not running in an IMS environment can update DB2 tables defined with DATA CAPTURE CHANGES. However, these changes are not propagated to IMS.

ANY  Allows subsystems to propagate some data with DPROP and other data with a different propagation program.

Tables that should only be updated by DB2 applications running in an IMS environment can be protected using the following methods:

•  Use the ENABLE parameter on BIND to specify a specific attachment facility through which updates to data propagation tables can be made.
•  Define a validation procedure for data propagation tables to define which plans can update those tables.
•  Allow update authority for data propagation tables to a group of authorization IDs that can only run in IMS.

Install parameter DPROP SUPPORT on panel DSNTIPO, or ZPARM EDPROP and CHGDC in DSN6SPRM.

Field Name: QWP4ENF

MAX TSPACE LOCK

The default (SYSTEM) for the LOCKMAX clause of the SQL statements CREATE TABLESPACE and ALTER TABLESPACE.

Install parameter LOCKS PER TABLE(SPACE) on panel DSNTIPJ, or ZPARM NUMLKTS in DSN6SPRM.
Field Name: QWP4LKTS

WAIT RETAIN LOCKS

Indicates whether a request is suspended until an incompatible retained lock becomes available.

This value is only significant in a data sharing environment. It indicates how long a transaction should wait for a lock on a resource if another DB2 in the data sharing group has failed and is holding an incompatible lock on that resource. Locks held by failed DB2 members are called retained locks.

This value is a multiplier that is applied to the connection’s normal timeout value. For example, if the retained lock multiplier is 2, then the timeout period for a call attachment connection that is waiting for a retained lock is twice the normal CAF timeout period. The default is 0, meaning applications do not wait for incompatible retained locks, the lock request is immediately rejected and the application receives a “resource unavailable” SQLCODE.

Install parameter RETAINED LOCK TIMEOUT on panel DSNTIPI, or ZPARM RETLWAIT in DSN6SPRM.

Field Name: QWP4WAIT

AUTO BIND

Indicates whether autobind is enabled. Values are:

YES  Allows automatic rebinding to be performed when a plan/package:
    • Was marked “invalid”.
    • Was bound on DB2 Vn, but is now running on DB2 Vn-1
    • After use on DB2 Vn-1 (as previously described), is later used again on DB2 Vn

NO Prevent DB2 from performing any automatic rebinding operations under any circumstances.

COEXIST

Allows automatic rebinding operation to be performed in a DB2 Data Sharing coexistence environment when the plan/package:

• Is marked “invalid” or
• Was last bound in DB2 Vn and is running on DB2 Vn-1

Install parameter AUTO BIND on panel DSNTIPO, or ZPARM ABIND in DSN6SPRM.

Field Name: QWP4ABN

ENABLE DB2 AUTH

Shows whether DB2 performs authorization checking.

When all authorization checking by DB2 is disabled, the GRANT statement is also disabled (granting every privilege to PUBLIC); this is not recommended.

Install parameter USE PROTECTION on panel DSNTIPP, or ZPARM AUTH in DSN6SPRM.

Field Name: QWP4AUTH
MAX APPL LOCKS

The maximum number of page or row locks that a single application can hold concurrently on all table spaces.

This includes locks on data pages, index pages, subpages, and rows that the program acquires when it accesses table spaces.

The limit applies to all table spaces defined with the LOCKSIZE PAGE, LOCKSIZE ROW, or LOCKSIZE ANY options. 0 means that there is no limit to the number of page and row locks a program can acquire.

DB2 assumes that 250 bytes of storage are required for each lock. If NO is specified for CROSS MEMORY, the value of this field has to take into account the available lock space. If referential constraints between tables is defined, the value of this field might need to be increased.

Install parameter LOCKS PER USER on panel DSNTIPJ, or ZPARM NUMLKUS in DSN6SPRM.

Field Name: QWP4LKUS

CACHE DYNAMIC SQL

Indicates whether prepared dynamic SQL statements are saved for later use by eligible application processes in the EDM pool.

Install parameter CACHE DYNAMIC SQL on panel DSNTIP8, or ZPARM CACHEDYN in DSN6SPRM.

Field Name: QWP4CDYN

EXPLAIN AUTOBIND

Indicates whether EXPLAIN processing occurs during automatic rebind.

YES means EXPLAIN processing happens during automatic rebind of a plan or package that has EXPLAIN(YES) as a bind option. If the PLAN_TABLE does not exist, automatic rebind continues, but there is no EXPLAIN output. Explain processing does not happen for a plan or package with EXPLAIN(NO).

Install parameter EXPLAIN PROCESSING on panel DSNTIPO, or ZPARM ABEXP in DSN6SPRM.

Field Name: QWP4ABX

AUTH CACHE SIZE

The size of the authorization cache to be used if no CACHESIZE is specified on the BIND PLAN subcommand.

The size of the cache is 32 bytes of overhead + (8 bytes of storage X number of concurrent users).

0 means authorization caching is not used.

Install parameter PLAN AUTH CACHE on panel DSNTIPP, or ZPARM AUTHCACH in DSN6SPRM.

Field Name: QWP4AUCA

REP READ U LOCK

Indicates whether the U (UPDATE) lock is used when using repeatable read (RR) or read stability (RS) isolation to access a table.
When YES, the U lock is used for an updated cursor with repeatable read or read stability.

When NO, the S lock is used for an updated cursor with repeatable read or read stability. If the cursor in the running applications includes the clause FOR UPDATE OF, but updates are infrequent, S locks generally provide better performance.

Install parameter U LOCK FOR RR/RS on panel DSNTIPI, or ZPARM RRULOCK in DSN6SPRM.

**Field Name:** QWP4RRU

**MAX KEPT DYN STMT**

Shows the total number of prepared dynamic SQL statements that are saved past a commit point.

0 means that prepared dynamic SQL statements are not saved past commit points.

Install parameter MAX KEPT DYN STMTS on panel DSNTIPE, or ZPARM MAXKEEPD in DSN6SPRM.

**Field Name:** QWP4MXKD

**HOP SITE AUTHORIZ.**

Indicates whose authorization is checked at a second server (sometimes called a hop site) when the requester is not DB2 for z/OS.

This option applies only when DB2 private protocol is used for the hop from the second to the third site.

When BOTH (default), the package owner’s authorization is checked for static SQL, and the runner’s authorization ID is checked for dynamic SQL.

When RUNNER, both static and dynamic SQL use the runner’s authorization.

Install parameter AUTH AT HOP SITE on panel DSNTIP5, ZPARM HOPAUTH in DSN6SPRM.

**Field Name:** QWP4HOP

**BIND NEW PACKAGE**

Shows whether BIND or BINDADD authority is required to BIND a new version of an existing package.

When BINDADD (default), only users with BINDADD system privilege can create a new package.

BIND users with BIND privilege for a package or collection can create a new version of an existing package when they bind it. This also allows users with PACKADM authority to add a new package or a new version of a package to a collection.

Install parameter BIND NEW PACKAGE on panel DSNTIPP, or ZPARM BINDNV in DSN6SPRM.

**Field Name:** QWP4BNVA

**CURRENT DEGREE**

Shows the default for the CURRENT DEGREE special register when no degree is explicitly set with SET CURRENT DEGREE.
IFCID 106 - Miscellaneous Installation Parameters

The default disables query parallelism.
Install parameter CURRENT DEGREE on panel DSNTIP8, or ZPARM CDSSRDEF in DSN6SPRM.

**Field Name:** QWP4CDEG

**IMS/BMP TIMEOUT**

The number of RESOURCE TIMEOUT units that an IMS BMP connection waits for a lock to be released.
The default value is 4, meaning that an IMS BMP connection can wait 4 times the resource timeout value for a resource.
Install parameter IMS BMP TIMEOUT on panel DSNTIPI, or ZPARM BMPTOUT in DSN6SPRM.

**Field Name:** QWP4WBMP

**TRACKER SITE**

Indicates whether this subsystem is a remote tracker site for another DB2 subsystem.
When YES, this is a tracker site.
A DB2 tracker site is a separate DB2 subsystem or data sharing group that exists solely for the purpose of keeping shadow copies of your primary site's data. No independent work can be run on the tracker site.
Install parameter TRACKER TYPE on panel DSNTIPO, or ZPARM TRKRSITE in DSN6SPRM.

**Field Name:** QWP4TRKR

**SORT POOL SIZE**

Indicates the amount of storage needed for the sort pool.
This can be the value calculated by the CLIST, based on input from previous panels, or the value entered in the Override column at installation time.
Install parameter SORT POOL SIZE on panel DSNTIPC, or ZPARM SRTPPOOL in DSN6SPRM.

**Field Name:** QWP4SPOL

**STATIC DESCRIBE**

Shows whether DB2 builds a DESCRIBE SQLDA when binding static SQL statements.
A DESCRIBE cannot be issued against a static SQL statement except:
- In a distributed environment, where DB2 for z/OS is the server and the requester supports extended dynamic SQL. In this instance, a DESCRIBE on an SQL statement in the extended dynamic package appears to DB2 as a DESCRIBE on a static SQL statement in the DB2 package.
- When an application uses a stored procedure result set, the application must allocate a cursor for that result set. The application can do this using a DESCRIBE CURSOR statement. The SQL statement actually described is the one with the cursor declared in the stored procedure. If that statement is static, a static SQL statement must be described.
When NO (default), DB2 does not generate a DESCRIBE SQLDA at BIND time for static SQL statements. If a DESCRIBE request is received at execution time, DB2 generates an error. However, if the describe request comes from a DESCRIBE CURSOR statement, DB2 satisfies the request but is only able to provide data type and length information. Column names are not provided.

When YES, DB2 generates a DESCRIBE SQLDA at BIND time so that DESCRIBE requests for static SQL can be satisfied during execution.

**Note:** You must rebind packages after this value has been set to YES.

This option increases the size of some packages because the DESCRIBE SQLDA is now stored with each statically-bound SQL SELECT statement.

Install parameter DESCRIBE FOR STATIC on panel DSNTIP4, or ZPARM DESCSTAT in DSN6SPRM.

**Field Name:** QWP4DSST

**IMS/DLI TIMEOUT**

The number of RESOURCE TIMEOUT units that a DL/I batch connection waits for a lock to be released.

The default value is 6, meaning that an IMS BMP connection can wait 4 times the resource timeout value for a resource.

Install parameter DL/I BATCH TIMEOUT on panel DSNTIPI, or ZPARM DLITOUT in DSN6SPRM.

**Field Name:** QWP4WDLI

**OPT HINTS ALLOWED**

Shows whether DB2 can use optimization hints from the PLAN_TABLE to influence the access paths used for certain queries.

Install parameter OPTIMIZATION HINTS on panel DSNTIP8, or ZPARM OPTHINTS in DSN6SPRM.

**Field Name:** QWP4HINT

**RIDPOOL SIZE (KB)**

The amount of storage needed for the RID pool.

This can be the value calculated by the CLIST, based on input from previous panels, or the value entered in the Override column at installation time.

When 0, DB2 does not use access paths or join methods that depend on RID pool storage.

Install parameter RID POOL SIZE on panel DSNTIPC, or ZPARM MAXRBLK in DSN6SPRM.

**Field Name:** QWP4RMAX

**PACK AUTH CACHE**

The amount of storage allocated for caching authorization information for all packages on this DB2 member.

32 KB hold about 375 collection-ID.package-IDs. The cache is stored in the DSN1DBM1 address space.
IFCID 106 - Miscellaneous Installation Parameters

Install parameter PACKAGE AUTH CACHE on panel DSNTIPP, or ZPARM CACHEPAC in DSN6SPRM.

Field Name: QWP4PAC

CONTR THREAD STOR

Indicates whether DB2 returns unused thread storage at commit. Possible values are:

YES  DB2 checks threads at commit points and periodically returns unused storage to the system.

NO   DB2 does not check threads at commit points and returns acquired storage on deallocation.

Install parameter CONTRACT THREAD STG on panel DSNTIPE, or ZPARM CONSTOR in DSN6SPRM.

Field Name: QWP4CONT

MAX DEG OF PARALL

Indicates the upper limit on the degree of parallelism for a parallel group.

This field has a value of 0. This means PARAMDEG is not set and DB2 can set a default maximum degree of parallelism based on the system configuration.

Install parameter MAX DEGREE on panel DSNTIP8, or ZPARM PARAMDEG in DSN6SPRM.

Field Name: QWP4MDEG

RTN AUTH W/O CAT

The amount of storage allocated for caching authorization information for all routines on this DB2 member.

Routines include stored procedures and user-defined functions.

32 KB hold about 380 schema.routine.type entries.

Install parameter ROUTINE AUTH CACHE on panel DSNTIPP, or ZPARM CACHERAC in DSN6SPRM.

Field Name: QWP4RAC

UPD PART KEY COLS

Indicates whether values in columns that participate in partitioning keys can be updated.

Possible values are YES, NO, or SAME. When SAME, updates are allowed only when the updated row remains in the same partition. The default value is YES.

Install parameter UPDATE PART KEY COLS on panel DSNTIP8, or ZPARM PARTKEYU in DSN6SPRM.

Field Name: QWP4PKYU

USE X LOCK

The locking method used when performing a searched UPDATE or DELETE.

When NO, DB2 uses an S or U lock when scanning for qualifying rows. For any qualifying rows or pages the lock is upgraded to an X lock before
performing the update or delete. For nonqualifying rows or pages the lock is released if using ISOLATION(CS). For ISOLATION(RS), or ISOLATION(RR), an S lock is retained on the rows or pages until the next commit point. This option is used to achieve higher rates of concurrency.

When YES, DB2 gets an X lock on qualifying rows or pages. For ISOLATION(CS), the lock is released if the rows or pages are not updated or deleted. For ISOLATION(RS) or ISOLATION(RR), an X lock is retained until the next commit point. This is beneficial in a data sharing environment when most or all searched updates and deletes use an index. The downside is that if searched updates or deletes result in a tablespace scan, the likelihood of timeouts and deadlocks greatly increases.

Install parameter X LOCK FOR SEARCHED U/D on panel DSNTIPI, or ZPARM XLKUPDLT in DSN6SPRM.

**Field Name:** QWP4XLUD

**EDM BEST FIT**

Shows the free chain search algorithm on systems with a large EDM pool (greater than 40 MB). Possible values are:

- **YES** Use a better fit algorithm.
- **NO** Use a first fit algorithm.

Install parameter LARGE EDM BETTER FIT on panel DSNTI8, or ZPARM EDMBFIT in DSN6SPRM.

**Field Name:** QWP4EBF

**STAR JOIN ENABL**

Star join enable indicator. Possible values are:

- **-1 (DISABLE)** Star join is disabled. This is the default.
- **0 (ENABLE)** Star join is enabled when the join meets the conditions described in the DB2 administration information for performance.
- **1** Star join is enabled without comparing the ratio of the fact-table cardinality to the cardinality of the largest dimension table. The table with the largest cardinality is the fact table.
- **n** This is the star join fact table and the largest dimension table ratio. The lowest ratio of the cardinality of the fact table compared to the cardinality of the largest dimension table for which star join is used. $2 < N <= 32768$.

Install parameter STAR JOIN QUERIES on panel DSNTI8, or ZPARM STARJOIN in DSN6SPRM.

**Background and Tuning Information**

This parameter allows you to set the star join ratio to increase or decrease the dimension table and fact table ratio rule according to application needs.

This parameter also allows you to disable star join if needed for performance reasons. The default is to allow star join if star join detection is successful.

Star join technique is only used when these conditions exist:

- At least two dimensions exist.
The join predicates are between the fact table and the dimension tables only. (No join predicates lie between the dimension tables.)

- The join predicates are equijoin predicates.
- No correlated subqueries cross dimensions.
- No cycles within the dimensions exist. This means that no predicate can reference more than one candidate dimension table with respect to the same column of the fact table.
- No outer join exists.
- The data type and length of the join predicates are the same.
- The fact table is larger than the dimension table.

**Field Name:** QWP4SJRT

**NPAGES THRESHOLD**

This parameter allows you to specify the optimizer threshold for qualifying a table as small.

-1 Every table qualifies as small.
0 No table qualifies as small (this is the default).
1 Only tables with zero pages qualify as small.
2 Tables with less than two pages qualify as small.
10 Tables with less than ten pages qualify as small.
502 Tables with less than 502 pages, and tables that have not had statistics collected qualify as small. For example, when NPAGES = -1.

DB2 parameter NPGTHRS in DSN6SPRM.

**Background and Tuning Information**

Tables can be populated using insert just prior to their use by queries and then cleared immediately on completion of the queries. These tables are permanent even though the data they contain is transient.

This can cause problems when RUNSTATS is run overnight, or at other times when these tables are empty. This gives the optimizer the false indication that these tables contain no data when in fact, the tables will contain data when the query executes. This causes the optimizer to pick an inefficient access path. Usually the optimizer chooses to do a table scan, which would be the most efficient access path if the table were truly empty. Because the table is not empty when the query executes, it would be more efficient to use matching index access.

With this parameter, you can force the optimizer to treat tables containing no data as small tables. For these tables, the optimizer will:

- Select a matching index access rather than a table space scan and non-matching index access.
- Select the index with the most matching columns when more than one index qualifies for matching index access.
- Select indexes with the same number of matching columns on cost.

**Field Name:** QWP4NPAG

**DBADM CREATE VIEW**
IFCID 106 - Miscellaneous Installation Parameters

Shows whether a DB2 administrator can create a view or alias for another user. Possible values are YES or NO. The default is NO.

Install parameter DBADM CREATE AUTH on panel DSNTIPP. ZPARM DBACRVW in macro DSN6SPRM.

Field Name: QWP4CRVW

MAX # LE TOKENS

The maximum number of LE tokens active at any time. When zero, no tokens are available.

A token is used each time one of the following is used: trigonometry functions, degrees, radians, rand, exp, power, log functions, upper, lower, translate.

Install parameter MAXIMUM LE TOKENS on panel DSNTIP7, or ZPARM LEMAX in DSN6SPRM.

Field Name: QWP4LEM

MAX EXT SERV TASK

Maximum number of extended service tasks.

Field Name: QWP4EST

CTR PCK HSH TABLE

The size of the control package hash table.

Field Name: QWP4KSIZ

PROJ Z INS THRESH

Project z insertion threshold.

Field Name: QWP4ZTN

MAX NOT FOUND-HASH

The maximum number of NOT FOUND hash records.

Field Name: QWP4KNFC

FIELD PROCES T BLK

The number of field procedures for the DESCRIBE TABLE block.

ZPARM SPRMFDP.

Field Name: QWP4FDP

MANAGE THREAD STO

Shows whether DB2 uses storage management to optimize the amount of working storage consumed by individual threads.

Install parameter MANAGE THREAD STORAGE on panel DSNTIPE, or ZPARM MINSTOR in DSN6SPRM.

For best performance, this parameter should be NO, meaning DB2 does not manage thread storage.

When YES, DB2 uses best fit algorithm to manage and assign thread storage. This can help on systems that have many long-running threads and that are constrained on DBM1 address space.

Field Name: QWP4MSTG
EVAL UNCOMMITTED

Shows whether stage 1 predicate evaluation during table access can proceed upon uncommitted data or not.

This applies to isolation levels of Read Stability and Cursor Stability only.

When NO (default), predicate evaluation occurs only on committed data (or on the application’s own uncommitted changes). NO ensures that all qualifying data is always included in the answer set.

When YES, predicate evaluation can occur upon uncommitted data. Only committed data is returned to the query. However, a decision can be made to omit a row from the answer set based on uncommitted data. Later, undo processing (statement rollback or statement failure) could cause the data to revert to a state that satisfies the predicate.

When YES, DB2 can request fewer locks than in previous versions when processing isolation level Read Stability and Cursor Stability queries. The number of locks avoided is related to the access path of the query, the number of rows evaluated when processing the stage 1 predicate of the query, and the number of those rows that are overflow rows. Specifically, for isolation level Read Stability and Cursor Stability queries, locks are avoided for rows that do not satisfy the stage 1 predicate, provided they are not overflow rows. Table access includes table space scans and index-to-data access, including ridlist-to-data access. For isolation Cursor Stability ridlist production, all row/page locking is avoided.

Install parameter EVALUATE UNCOMMITTED on panel DSNTIP8, or ZPARM EVALUNC in DSN6SPRM.

Field Name: QWP4EVUN

STATISTICS ROLLUP

Shows whether RUNSTATS utility aggregates the partition level statistics, even though some parts may not contain data.

This should be YES for DB2 systems that have large partitioned table spaces, index spaces, or both. This enables the aggregation of part level statistics and helps the optimizer to choose a better access path.

Install parameter STATISTICS ROLLUP on panel DSNTIPO, or ZPARM STATROLL in DSN6SPRM.

Field Name: QWP4STRL

STATISTICS HIST

Shows which inserts and updates are recorded in catalog history tables.

The report can show the following values:

N / NONE
  Changes in the catalog are not recorded. This is the default.

A / ALL
  All inserts and updates in the catalog are recorded.

P / ACCESSPATH
  All inserts and updates to access path related catalog statistics are recorded.
S / SPACE

All inserts and updates to space related catalog statistics are recorded.

Install parameter STATISTICS HISTORY on panel DSNTIPO, or ZPARM STATHIST in DSN6SPRM.

Field Name: QWP4STHT

SUPPRESS SOFT ERR

Shows whether the recording of errors, such as invalid decimal data and arithmetic exceptions, in the operating system data set SYS1.LOGREC is suppressed.

When YES, these exceptions are not recorded in the LOGREC data set.

Install parameter SUPPRESS SOFT ERRORS on panel DSNTIPM or ZPARM SUPERRS in DSN6SPRM.

Field Name: QWP4SAE

REAL TIME STATS

The time interval that DB2 waits before it attempts to write out page set statistics to the real-time statistics tables. This value is between 1 and 65535 minutes.

Install parameter REAL TIME STATS on panel DSNTIPO, or ZPARM STATSINT in DSN6SPRM.

Field Name: QWP4INTE

EDM STATEMENT CACHE

The size of the statement cache that can be used by the Environmental Descriptor Manager (EDM). This value is used at DB2 startup time as the minimum value. You can increase and subsequently decrease this value with the SET SYSPARM command. This value cannot be decreased below the value that is specified at DB2 startup. The CLIST calculates a statement cache size. This storage pool is located above the 2 GB bar.

The value used at DB2 startup time is either calculated by the CLIST based on input from other installation information or an override value.

For record trace, this value is shown in bytes. For other reports, the value is shown in kilobytes.

Install parameter EDM STATEMENT CACHE on panel DSNTIPC, or ZPARM EDMSTMTC in DSN6SPRM.

Field Name: QWP4ESTC

EDM DBD CACHE

The minimum size of the DBD cache that can be used by the Environmental Descriptor Manager (EDM). This value is used at DB2 startup time as the minimum value. You can increase and subsequently decrease the value with the SET SYSPARM command. This value cannot be decreased below the value that is specified at DB2 startup. This storage pool is located above the 2 GB bar. The CLIST calculates the DBD cache size.

The value used at DB2 startup time is either calculated by the CLIST based on input from other installation information or an override value.
IFCID 106 - Miscellaneous Installation Parameters

Install parameter EDM DBD CACHE on panel DSNTIPC, or ZPARM EDMDBDC in DSN6SPRM.

Field Name: QWP4EDBC

STAR JOIN THRESH
The minimum number of tables in the star schema query block, including the fact table, dimensions tables, and snowflake tables. This value is considered only if the subsystem parameter STARJOIN qualifies the query for star join.

Possible values are:

0       Star join is disabled. This is the default.
1, 2, or 3     Star join is always considered.
4 through 255   Star join is considered if the query block has at least the specified number of tables.
256 and greater Star join is never considered.

DB2 parameter SJTABLES in DSN6SPRM.

Background and Tuning Information
Although star join can reduce bind time significantly it does not provide optimal performance in all cases. Performance of star join depends on a number of factors such as the available indexes on the fact table, the cluster ratio of the indexes, and the selectivity of rows through local and join predicates. Follow these general guidelines for setting the value of SJTABLES:

• If you have star schema queries with less than 10 tables and you want to make the star join method applicable to all qualified queries, set the value of SJTABLES to a low number, such as 5.
• If you have some star schema queries that are not necessarily suitable for star join but want to use star join for relatively large queries, use the default. The star join method will be considered for all qualified queries that have 10 or more tables.
• If you have star schema queries but normally do not want to use star join, you could increase SJTABLES, say to 15. This will greatly cut the bind time for large queries and avoid a potential bind time SQL return code -101 for large qualified queries.

Field Name: QWP4SJTB

ZOSMETRICS
YES indicates that gathering of z/OS metrics using the RMF interface is enabled. ZPARM ZOSMETRICS in DSN6SPRM.

Field Name: QWP4METE

LONG RUNNING READ
Shows the number of minutes that a read claim can be held by an agent before DB2 reports it as a long-running reader. Valid values are 0 (default) through 1439.
IFCID 106 - Miscellaneous Installation Parameters

Install parameter LONG-RUNNING READER on installation panel DSNTIPE, or ZPARM LRDRTHLD in DSN6SYSP.

**Field Name:** QWP4LRTH

**TEMP UNIT NAME**
Shows the device type or unit name for allocating temporary data sets. It is the direct access or disk unit name used for the precompiler, compiler, assembler, sort, linkage editor, and utility work-files in the tailored jobs and CLISTs.

It can be any device type acceptable to the DYNALLOC parameter of the SORT or OPTION options for DFSORT.

The default is SYSDA.

Install parameter TEMPORARY UNIT NAME on DSNTIIPA2, or ZPARM VOLTDEVT in DSN6SPRM.

**Field Name:** QWP4VDTY

**MIN DIVIDE SCALE**
The minimum scale for the result of a decimal division. The values for this parameter are none (the default), 3, or 6. If 3 or 6 is specified, this parameter overrides the DECDIV3 parameter.

**Field Name:** QWP4MDSC

**CUR MAINT TYPE**
Shows the default special register for the CURRENT MAINTAINED TABLE TYPES FOR OPTIMIZATION statement when no value is explicitly set. Possible values are:

- **ALL**
- **NONE**
- **SYSTEM** (default)
- **USER**

The default allows query rewrite using system-maintained materialized query tables (SYSTEM) when CURRENT REFRESH AGE is set to ANY. When USER, query rewrite is done using user-maintained materialized query tables when CURRENT REFRESH AGE is set to ANY. ALL means that query rewrite uses both system-maintained and user- maintained materialized query tables.

Install parameter CURRENT MAINT TYPES on panel DSNTIP8, or ZPARM MAINTYPE in DSN6SPRM.

**Field Name:** QWP4MNTY

**PAD IDX BY DEFLT**
Shows whether new indexes are be padded by default.

- **YES** indicates that a new index is padded unless the NOT PADDED option is specified on the CREATE INDEX statement.
- The default value, **NO**, indicates that a new index is not padded unless the PADDED option is specified on the CREATE INDEX statement.

Install parameter PAD INDEXES BY DEFAULT on installation panel DSNTIPE, or ZPARM PADIX in DSN6SPRM.

**Field Name:** QWP4PDIX
IFCID 106 - Miscellaneous Installation Parameters

CUR REFRESH AGE
Shows the default for the CURRENT REFRESH AGE special register deferred materialized query tables.
Install parameter CURRENT REFRESH AGE on panel DSNTIP8, or ZPARM REFRESHAGE in DSN6SPRM.
Field Name: QWP4RFSH

FREE CACHED STMTS
Indicates whether DB2 can free statements from the local dynamic statement cache to relieve storage constraints below the 2 GB bar. This parameter applies only for packages or plans that are bound with KEYPDYNAMIC(YES). Possible values are:
0  DB2 cannot free statements from the local cache
1  DB2 can free statements from the local cache
DB2 parameter CACHEDYN_FREELOCAL in DSN6SPRM.
Field Name: QWP4FRLC

MAX OPEN CURSORS
Shows the maximum number of cursors, including allocated cursors, that are open at a given DB2 site per thread. RDS keeps a total of currently open cursors. If an application attempts to open a thread after the maximum is reached, the statement will fail.
In a data sharing group, this parameter is shown at member scope.
Install parameter MAX OPEN CURSORS on panel DSNTIPX, or ZPARM MAX_NUM_CUR in DSN6SPRM.
Field Name: QWP4MXNC

MAX STORED PROCS
Shows the maximum number of stored procedures per thread. If an application attempts to call a stored procedure after this is reached, the statement will fail. In a data sharing group, this parameter is shown as member scope.
Install parameter MAX STORED PROCS on panel DSNTIPX, or ZPARM MAX_ST_PROC in DSN6SPRM.
Field Name: QWP4MXSP

MAX DATA CACHING
The maximum amount of virtual memory in megabytes (MB) that is allocated for data caching.
Install parameter MAX DATA CACHING on panel DSNTIP8, or ZPARM MXDTCACHE in DSN6SPRM.
Field Name: QWP4MXDC

ONL ZPARM TYPE
The type of DB2 system parameter changed by the last SET SYSPARM statement.
Field Name: QWP4OZTP

ONL ZPARM USER ID
IFCID 106 - Miscellaneous Installation Parameters

The user ID that made the last online change to DB2 system settings.

**Field Name:** QWP4OZUS

**ONL ZPARM CORID**

The correlation ID of the online application that made the last change to DB2 system settings.

**Field Name:** QWP4OZCI

**ONL ZPARM TIME**

Time of the last online change made to DB2 system settings.

**Field Name:** QWP4OZTM

**DEFAULT INDEX TYPE**

Default index type.

**Field Name:** QWP4DXTP

**REL. CURSOR HOLD LOCKS**

Indicates whether, at commit time, DB2 should release a data page or row lock on which a cursor defined WITH HOLD is positioned.

This lock is not necessary for maintaining cursor position.

YES indicates that DB2 releases this data page or row lock after a COMMIT is issued for cursors defined WITH HOLD. This can improve concurrency.

NO indicates that DB2 holds the data page or row lock for WITH HOLD cursors after the COMMIT. This option is provided to allow existing applications, which rely on this lock to continue to work correctly.

Otherwise this field is left blank.

Install parameter RELEASE CURSOR HOLD LOCKS on panel DSNTIP8, or ZPARM RELCURHL in DSN6SPRM.

This DB2 field has been deprecated with DB2 9.

**Field Name:** QWP4RCHL

**3990 CACHE**

Indicates whether DB2 prefetch uses sequential mode to read cached data from a 3990 controller. When BYPASS (default), DB2 prefetch bypasses the cache.

When SEQ, DB2 prefetch uses sequential access for read activity. There is a performance benefit using SEQ with DFSMS or DFP controls with newer 3990 caches.

Install parameter SEQUENTIAL CACHE on panel DSNTIPE, or ZPARM SEQCACH in DSN6SPRM.

**Field Name:** QWP4SCAC

**RID POOL SIZE**

The amount of storage needed for the RID pool.

This can be the value calculated by the CLIST, based on input from previous panels, or the value entered in the Override column at installation time.
When 0, DB2 does not use access paths or join methods that depend on RID pool storage.

Install parameter RID POOL SIZE on panel DSNTIPC, or ZPARM MAXRBLK in DSN6SPRM.

**Field Name:** QWP4RMAX

**UTILITY CACHE OPT**

Shows whether utilities that scan a nonpartitioned index followed by an update of a subset of the pages in the index allow data to remain in 3990 cache longer when reading data.

Install parameter UTILITY CACHE OPTION on panel DSNTIPE, or ZPARM SEQPRES in DSN6PRM.

**Field Name:** QWP4PST

**VARCHAR FROM INDEX**

Indicates whether DB2 can return data from an index key for a variable length column.

**Field Name:** QWP4VCFC

**EDM POOL DSP SIZE**

Size of EDM pool data space, in kilobytes, calculated during installation. Valid values are 0 through 2097152 kilobytes. If dynamic statement cache is disabled, CACHEDYN is NO and the value shown here is 0. If dynamic statement cache is enabled, CACHEDYN is YES and the value shown here is 1048576.

Install parameter EDMPOOL DATA SPACE SIZE on panel DSNTIPC or ZPARM EDMDSPAC in DSN6PRM.

**Field Name:** QWP4EDDS

**SMS DC NAME TS**

SMS data class for data table spaces. The data class name is a string of one to eight characters. The default is an empty string, which means that the SMS cluster is defined without the DATACLASS parameter.

When a valid data class name is specified, the SMS cluster is specified with the DATACLASS parameter using the name specified. If the name is not valid, SMS returns an error.

DB2 parameter SMSDCFL in DSN6SPRM.

**Field Name:** QWP4DCFS

**SMS DC NAME IS**

SMS data class for index table spaces. The data class name is a string of one to eight characters. The default is an empty string, which means that the SMS cluster is defined without the DATACLASS parameter.

When a valid data class name is specified, the SMS cluster is specified with the DATACLASS parameter using the name specified. If the name is not valid, SMS returns an error.

DB2 parameter SMSDCIX in DSN6SPRM.

**Field Name:** QWP4DCIX

**MAX EDM POOL SIZE**
Maximum size in kilobytes to which the data space for EDM can be increased. Valid values are 0-2097152. If dynamic statement cache is disabled, CACHEDYN is NO and the default value here is 0. If dynamic statement cache is enabled, CACHEDYN is YES and the default value here is 1048576.

Installation parameter SPRMEDMX in panel DSNTIPC, or ZPARM EDMDSMAX in DSN6SPRM.

Field Name: QWP4EDMX

MAXREADS ST AREA

Maximum storage for allocated reads.

This DB2 field has been deprecated with DB2 9.

Field Name: QWP4IFS

COST-BASED PARAL SORT

Indicates whether parallel sort for multi-table sort for composite is active.

Field Name: QWP4OPSE

OUTER JOIN PERFORM ENH

Indicates whether outer join performance enhancements are enabled.

DB2 parameter OJPERFEH in DSN6SPRM.

Field Name: QWP4OJEH

MAX TEMPS STORAGE

The maximum amount of temporary storage in megabytes (MB) for each agent.

Install parameter MAX TEMP STORAGE on panel DSNTIP6 or ZPARM MAXTEMPS in DSNTIP9.

Field Name: QWP4WFAL

MAX CONC AUTOBIND

The maximum number of package requests that can be processed simultaneously.

DB2 parameter MAX_CONCURRENT_PKG_OPS in DSN6SPRM.

Field Name: QWP4MXAB

EDM SKELE POOL SIZE

The minimum size of the EDM pool for skeleton package and skeleton cursor tables. For record trace, this value is shown in bytes. For other reports, the value is shown in kilobytes.

Install parameter EDM SKELETON POOL SIZE on panel DSNTIPC or ZPARM EDM_SKELETON_POOL in DSN6SPRM.

Field Name: QWP4SKLC

ADMN SCHED JCLPROC

The name of the JCL procedure for starting the DB2 administrative scheduler task address space.

DB2 parameter ADMTPROC in DSN6SPRM.
IFCID 106 - Miscellaneous Installation Parameters

Field Name: QWP4ADMT

SYS-LEVEL BACKUP

Shows if RECOVER uses system level backups as the recovery base.

Install parameter SYSTEM-LEVEL BACKUPS on installation panel DSNTIP6, or ZPARM SYSTEM_LEVEL_BACKUPS in DSN6SPRM.

Field Name: QWP4SLBU

RESTORE/RECOVER

If YES, the system-level backup that is the recovery base, is from a dump on tape. Otherwise NO is shown.

Install parameter RESTORE/RECOVER on installation panel DSNTIP6, or ZPARM RESTORE_RECOVER_FROMDUMP in DSN6SPRM.

Field Name: QWP4RRFD

DUMP CLASS NAME

The name of the DFSMSHSM dump class used by the restore system utility to restore from a system-level backup that has been dumped to tape.

Install parameter DUMP CLASS NAME on installation panel DSNTIP6, or ZPARM UTILS_DUMP_CLASS_NAME in DSN6SPRM.

Field Name: QWP4RSDC

MAX TAPE UNITS

The maximum number of tape units or tape drives that the restore system utility can use to restore from a system-level backup that has been dumped to tape.

Install parameter MAXIMUM TAPE UNITS on installation panel DSNTIP6, or ZPARM RESTORE_TAPEUNITS in DSN6SPRM.

Field Name: QWP4RSMT

INDEX I/O PARALL

The enablement of the index I/O parallelism ZPARM.

Field Name: QWP4IIOP

PLANMGMT

Shows if and how access path information is stored in the repository.

Possible values are:

- O: On
- F: Off
- B: Basic
- E: Extended

Field Name: QWP4PMGT

PLANMGMTSCOPE

Controls which queries are populated in the access path repository (ZPARM parameter PLANMGMTSCOPE). Possible values are:

- A: ALL: Includes static and dynamic SQL queries.
- S: STATIC: Includes static SQL queries only. This is the default.
**IFCID 106 - Miscellaneous Installation Parameters**

**D**  
**DYNAMIC:** Includes dynamic SQL queries only.  
**Field Name:** QWP4PMSM

**REVOKE DEP PRIVIL**  
Include dependent privileges on REVOKE. Possible values are:  

- **Y**  
  If INCLUDING DEPENDENT PRIVILEGES is enforced.  
- **N**  
  If NOT INCLUDING DEPENDENT PRIVILEGES is enforced.  
- **S**  
  If specified in a REVOKE statement.  
**Field Name:** QWP4RVDPR

**SEPARATE SECURITY**  
Separate security tasks. Possible values are:  

- **Y**  
  SYSADM/SYSCTRL cannot GRANT/REVOKE  
- **N**  
  SYSADM/SYSCTRL can GRANT/REVOKE  
**Field Name:** QWP4SEPSD

**SECADM1_TYPE**  
Security administrator 1 type. Possible values are:  

- **'**  
  *Blank* indicates that the authorization ID (AUTH ID) is used.  
- **'L'**  
  Indicates that ROLE is used.  
This field corresponds to field SEC ADMIN 1 TYPE on installation panel DSNTIPP1, or ZPARM SECADM1_TYPE in DSN6SPRM.  
**Field Name:** QWP4SECA1_TYPE

**SECADM2_TYPE**  
Security administrator 2 type. Possible values are:  

- **'blank**  
  Indicates that the authorization ID (AUTH ID) is used.  
- **'L'**  
  Indicates that ROLE is used.  
This field corresponds to field SEC ADMIN 2 TYPE on installation panel DSNTIPP1, or ZPARM SECADM2_TYPE in DSN6SPRM.  
**Field Name:** QWP4SECA2_TYPE

**MAX TEMP RID**  
The maximum number of RID blocks of temporary storage in the Workfile database that a single RID list can use at any point in time. This field corresponds to field MAX TEMP RID on installation panel DSNTIP9. The ZPARM name is MAXTEMPS_RID.  
It can have the following values:  

- **-1**  
  if MAXTEMPS_RID=NONE  
- **0**  
  if MAXTEMPS_RID=NOLIMIT  
- **1 to 329166** otherwise  
**Field Name:** QWP4WFRD

**SECADM1 ID**  
Security administrator 1.  
**Field Name:** QWP4SECA1
SEADM2 ID

Security administrator 2.

Field Name: QWP4SECA2

SKIP UNCOMM INS

YES indicates that uncommitted inserts are treated as if they have not yet been executed. The ZPARM name is SKIPUNCI.

Field Name: QWP4SKUI

GET ACCEL ARCHIVE

Determines the default value that is to be used for the CURRENT GET_ACCEL_ARCHIVE special register:

NO Indicates that if a table is archived in an accelerator server, and a query references that table, the query does not use the data that is archived.

YES Indicates that if a table is archived in an accelerator server, and a query references that table, the query uses the data that is archived.

ZPARM name GET_ACCEL_ARCHIVE in macro DSN6SPRM.

Field Name: QWP4CGAA

QUERY ACCEL OPT

Specifies additional types of SQL queries that are eligible for acceleration.

NONE Indicates that no additional types of SQL queries are eligible. Therefore, the types of queries that are described in the other available values for this parameter are not eligible for acceleration. This is the default value.

1 Indicates that queries that include data that is encoded with the EBCDIC mixed or graphic encoding schemes are eligible for acceleration.

2 Indicates that an INSERT with SELECT statement is eligible for acceleration. However, only the SELECT operation of the query is processed by the accelerator server.

3 Indicates that queries that contain built-in functions for which DB2 processes each byte of the input string, rather than each character of the input string, can run on an accelerator server.

4 The queries that reference an expression with a DATE data type that uses a LOCAL format are not blocked from executing on IDAA. IDAA will use the dd/mm/yyyy format to interpret the input and output date value. Specify option 4 only when you also specify LOCAL as the setting for the DSNHDECP.DATE parameter and your LOCAL date exit defines the specific dd/mm/yyyy date format. Otherwise, queries may return unpredictable results.

ZPARM name QUERY_ACCEL_OPTIONS in macro DSN6SPRM.

Field Name: QWP4QACO

CUR QUERY ACCEL
IFCID 106 - Miscellaneous Installation Parameters

Determines the default value that is to be used for the CURRENT QUERY ACCELERATION special register. Possible values are:

NONE
Indicates that no query acceleration is done. This is the default value.

ENABLE
Indicates that queries are accelerated only if DB2 determines that it is advantageous to do so. If there is an accelerator failure while a query is running, or the accelerator returns an error, DB2 returns a negative SQLCODE to the application.

ENABLE_WITH_FAILBACK
Indicates that queries are accelerated only if DB2 determines that it is advantageous to do so. If the accelerator returns an error during the PREPARE or first OPEN for the query, DB2 executes the query without the accelerator. If the accelerator returns an error during a FETCH or a subsequent OPEN, DB2 returns the error to the user, and does not execute the query.

ELIGIBLE
Indicates that queries are accelerated if they are eligible for acceleration. DB2 does not use cost information to determine whether to accelerate the queries. Queries that are not eligible for acceleration are executed by DB2. If there is an accelerator failure while a query is running, or the accelerator returns an error, DB2 returns a negative SQLCODE to the application.

ALL
Indicates that queries are accelerated if they are eligible for acceleration. DB2 does not use cost information to determine whether to accelerate the queries. Queries that are not eligible for acceleration are not executed by DB2, and an SQL error is returned. If there is an accelerator failure while a query is running, or the accelerator returns an error, DB2 returns a negative SQLCODE to the application.

ZPARM name QUERY_ACCELERATION in DSN6SPRM.

Field Name: QWP4CQAC

DDL TIMEOUT FACT
Shows the time out factor of the SQL data definition. The time out value is the product of this value and the IRLMRWT value.

ZPARM name DDLTOX in DSN6SPRM.

Field Name: QWP4DDLTO

LMT CONV PART TAB
Shows whether to include all columns in the partitioning key during conversion from index-controlled partitioning to table-controlled partitioning:

NO Includes all columns

YES Includes trailing columns only if they affect partitioning

This field corresponds to field EXCLUDE PART KEY ELEMENTS on installation panel DSNTIP71. The ZPARM name is IX_TB_PART_CONV_EXCLUDE in DSN6SPRM.
IFCID 106 - Miscellaneous Installation Parameters

Field Name: QWP4XPKE
MAX UTIL PARALL
The maximum degree of utility parallelism.
Field Name: QWP4UMD
ACCEL STARTUP OPT
Specifies whether to enable accelerator servers. Possible values are:
AUTO
   Enable and start accelerator servers.
COMMAND
   Enable but do not start accelerator servers.
NO
   Do not enable accelerator servers.
This field corresponds to field ACCEL STARTUP on installation panel DSNTIP81. ZPARM name is ACCEL in DSN6SPRM.
Field Name: QWP4ACCS
REORG IGN FREESP
YES indicates that REORG tablespace does not use the PCTFREE and FREEPAGE values when it reloads data rows into a partition-by-growth (PBG) table space if:
   • A subset of the partitions is reorganized.
   • The associated table contains LOB columns that cause a REORG AUX NO REQUEST to fail.
This field corresponds to field REORG IGNORE FREESPACE in installation panel DSNTIP61. ZPARM name is REORG_IGNORE_FREESPACE in DSN6SPRM.
Field Name: QWP4RIFS
MULT INDEX ACCESS
Specifies whether to enable or disable multiple index access for queries that have subquery predicates:
NO
   Disables multiple index access for queries.
YES
   Enables multiple index access for queries.
The ZPARM name is SUBQ_MIDX in DSN6SPRM.
Field Name: QWP4SQMX
REORG SORT NPSI
Specifies the default method of building a non-partitioned secondary index during the REORG tablespace part. This setting is used when the SORTNPSI keyword is not specified in a utility control statement.
Possible values are:
   • Auto
   • Disable
   • Enable
This field corresponds to field REORG PART SORT NPSI in installation panel DSNTIP61. The ZPARM name is REORG_PART_SORT_NPSI in DSN6SPRM.
IFCID 106 - Miscellaneous Installation Parameters

Field Name: QWP4RPSN

REORG TBSPC LIST

- Specifies the default value for the REORG TABLESPACE PARALLEL option.
- Parallel
- Serial

The ZPARM name is REORG_LIST_PROCESSING in DSN6SPRM.

Field Name: QWP4RLPR

OPT 1 ROW-NO SORT

- YES indicates that sort access paths are disabled (if possible) if OPTIMIZE for 1 row with a query is used. If a possible access path avoids a SORT, DB2 chooses this access path.

The ZPARM name is OPT1ROWBLOCKSORT in DSN6SPRM.

Field Name: QWP4O1RBS

AUTH EXIT CHECK

- Specifies whether the DB2 authorization ID or the RACF primary authorization ID is to be used for authorization checks, when the access control authorization exit is active:

Primary

- DB2 provides:
  - The ACEE of the package owner to perform statement authorization checks during AUTOMATIC REBIND, BIND, and REBIND processing
  - The ACEE of the package owner, routine definer, or routine invoker, as determined by the dynamic rules behavior for dynamic SQL authorization checking, when a DYNAMICRULES BIND option value other than run is in effect.

The access control authorization exit uses the ACEE for the XAPLUCHK authorization ID field to perform the authorization. The authorization ID in XAPLUCHK must be defined as a RACF user and must have the privileges required to execute the SQL statements in the package.

DB2

- DB2 provides the ACEE of the primary authorization ID for performing all authorization checks. The primary authorization ID must have the privileges required to execute the SQL statements in the package. This field corresponds to field "RACF AUTH CHECK" on installation panel DSNTIIP. ZPARM name is RACF_AUTHCHECK in DSN6SPRM.

Field Name: QWP4RACK

OBJ CREATE FORMAT

- Creates new table spaces and indexes in the following log record format:

EXTENDED

- Creates new table spaces and indexes in extended log record format.

BASIC

- Creates new table spaces and indexes in basic log record format.
IFCID 106 - Miscellaneous Installation Parameters

**Field Name:** QWP4OBCF

**UTIL OBJ CONVERS**

This field can have the following values:

- **NONE** (QWP4UTOC1=0 and QWP4UTOC2=0)
  
  No conversion is performed. This option is the default setting of this parameter. NONE is allowed regardless of the OBJECT CREATE FORMAT setting.

- **BASIC** (QWP4UTOC1=1 and QWP4UTOC2=0)
  
  Existing table spaces and indexes that use extended 10-byte page format are converted to basic 6-byte page format. BASIC is allowed only if the OBJECT CREATE FORMAT field is also set to BASIC.

- **EXTENDED** (QWP4UTOC1=0 and QWP4UTOC2=1)
  
  Existing table spaces and indexes that use 6-byte page format are converted to extended 10-byte page format. EXTENDED is allowed only if the OBJECT CREATE FORMAT field is also set to EXTENDED.

- **NO BASIC** (QWP4UTOC1=1 and QWP4UTOC2=1)
  
  Prevents the conversion of table spaces and indexes in extended page format to basic page format and disallows a utility that accepts the RBA_LRSN_CONVERSION utility keyword from running on an object in basic page format unless it converts it to extended page format. This setting is permitted only when OBJECT_CREATE_FORMAT=EXTENDED is set.

The ZPARM name is UTILITY_OBJECT_CONVERSION in DSN6SPRM.

**Field Name:** RT0106OC

**PKG RELEASE COMMIT**

YES indicates that the following operations on a package that are bound with RELEASE(DEALLOCATE) are permitted while the package is active and allocated by DB2:

- BIND and REBIND requests, including AUTOMATIC REBIND
- Data definition language changes to objects that are statically referenced by the package

The ZPARM name is PKGREL_COMMIT in DSN6SPRM.

**Field Name:** QWP4PKRC

**REORG DROP PARTS**

If YES, REORG completes, REORG drops empty, and trailing partitions are set in a PARTITION-BY-GROWTH table space.

This field corresponds to field REORG DROP PBG PARTS on INSTALLATION panel DSNTIP61. The ZPARM name is REORG_DROP_PBG_PARTS in DSN6SPRM.

**Field Name:** QWP4RPBG

**TEMPLATE TIME**

Specifies the default setting for the TIME option of the template utility control statement. Possible values are:

- UTC (utility control)
- Local
IFCID 106 - Miscellaneous Installation Parameters

Field Name: QWP4TPTM

AUTHEX CACHE REF

Specifies whether the package authorization cache, routine authorization cache, and dynamic statement cache entries are refreshed when an access control authorization exit is active, and the user profile is changed in RACF. Possible values are:

• All
• None

Field Name: QWP4AECR

SPT01 MAX LENGTH

The maximum length in bytes of LOB columns in the SPT01 directory space that are maintained in the base table. This field corresponds to field SPT01 INLINE LENGTH on installation panel DSNTIPA2. The ZPARM name is SPT01_INLINE_LENGTH in DSN6SPRM.

Field Name: QWP4S1IL

REORG MAPPING DB

The default database in which REORG TABLESPACE SHRLEVEL change implicitly creates the mapping table. This field corresponds to field RECORD MAPPING DB on installation panel DSNTIP61. The ZPARM name is RECORD_MAPPING_TABLE in DSN6SPRM.

Field Name: QWP4RMDB

MAX IN-MEM SORT

The maximum amount of storage in kilobytes to allocate for sorting the results of each query that contains the order by clause, the group by clause, or both. This field corresponds to field MAX IN-MEMORY SORT SIZE in installation panel DSNTIPC. The ZPARM name is MAXSORT_IN_MEMORY in DSN6SPRM.

Field Name: QWP4MIMTS

IDX CLEANUP THRDS

The maximum number of threads that can be created to clean up pseudo-deleted index entries on a data sharing member of a subsystem. This field corresponds to field INDEX CLEANUP THREADS on installation panel DSNTIPE1. The ZPARM name is INDEX_CLEANUP_THREADS in DSN6SPRM.

Field Name: QWP4IXCU

MAX PARA DEG DPSI

The maximum degree of parallelism for a parallel group in which a data-partitioned secondary index is used to control parallelism. This field corresponds to field MAX DEGREE FOR DPSI on installation panel DSNTIP81. The ZPARM name is PARAMDEG_DPSI in DSN6SPRM.
IFCID 106 - Miscellaneous Installation Parameters

Field Name: QWP4DEGD

APPL COMPAT

Specifies the DB2 level for downward compatibility with applications. The ZPARM name is APPLCOMPAT in DSN6SPRM.

Field Name: QWP4APCO_VAR

STATIST FEEDBACK

Specifies the scope of SQL statements for which DB2 is to recommend statistics. Possible values are:
- All
- Dynamic
- None
- Static

This value corresponds to field STATISTICS FEEDBACK on installation panel DSNTIPO. The ZPARM name is STATFDBK_SCOPE in DSN6SPRM.

Field Name: QWP4SFBS

LIKE BLANK INSIGN

YES indicates that blanks are not significant when DB2 applies the LIKE predicate to a string. Blanks are significant in DB2 10 or 9.

This setting corresponds to field LIKE BLANK INSIGNIFICANT on installation panel DSNTIP41. The ZPARM name is LIKE_BLANK_INSIGNIFICANT in DSN6SPRM.

Field Name: QWP4LBIN

PCTFREE UPDATE

Specifies the default percentage of each page that DB2 leaves as free space in a table space when a table in this table space is populated. This value applies only to table spaces whose definitions do not include PCTFREE and for UPDATE.

This value corresponds to field PERCENT FREE FOR UPDATE on installation panel DSNTIP71. The ZPARM name is PCTFREE_UPD in DSN6SPRM.

Field Name: QWP4PFUP

WF DB AGNT THRESH

Specifies the percentage of space that is used in the Workfile Database by a single agent when DB2 issues a warning message.

This value corresponds to field AGENT LEVEL THRESHOLD on installation panel DSNTIP91. The ZPARM name is WFSTGUSE_AGENT_THRESHOLD in DSN6SPRM.

Field Name: QWP4WFSAT

WF DB SYS THRESH

Specifies the percentage of space that is used in the Workfile Database by all agents in a DB2 subsystem or data sharing member when DB2 issues a warning message.
IFCID 106 - Miscellaneous Installation Parameters

This value corresponds to field SYSTEM LEVEL THRESHOLD on installation panel DSNTIP91. The ZPARM name is WFSTGUSE_SYSTEM_THRESHOLD in DSN6SPRM.

Field Name: QWP4WFSST

D_STMT CACHE STOR

Specifies the number of gigabytes of storage that DB2 allocates for hashing entries in the dynamic statement cache. This parameter can avoid storage shortages for long-running threads. The storage is allocated above the bar.

The ZPARM name is CACHE_DEP_TRACK_STOR_LIM in DSN6SPRM.

Field Name: QWP4CDTSL

FCOPY DEFLT TEMPL

Specifies the default setting of the FCCOPYDDN subsystem parameter for the COPY, LOAD, REBUILD INDEX, REORG INDEX, and REORG TABLESPACE utility control statements when the FLASHCOPY parameter is YES or CONSISTENT. FCCOPYDDN specifies a DB2 utility template data-set name expression that is used to derive the copy data-set name that is allocated by the utility during operation.

This field corresponds to field DEFAULT TEMPLATE on installation panel DSNTIP6. The ZPARM name is FCOPYDDN in DSN6SPRM.

Field Name: QWP4FCCD
IFCID 106 - Stored Procedures Parameters

This topic shows detailed information about “Record Trace - IFCID 106 - Stored Procedures Parameters”.

Record trace - IFCID 106 - Stored Procedures Parameters

The field labels shown in the following sample layout of “Record Trace - IFCID 106 - Stored Procedures Parameters” are described in the following section.

STORED PROCEDURES PARAMETERS

MVS PROCEDURE NAME

The name of the MVS JCL procedure used to start the DB2 stored procedures address space.

Install parameter DB2 PROC NAME on panel DSNTIPX, or ZPARM STORPROC in DSN6SYSP.

Field Name: QWP1SPPN

ALLOWABLE ABENDS

The number of times a stored procedure is allowed to terminate abnormally, after which SQL CALL statements for the stored procedure are rejected.

Install parameter MAX ABEND COUNT on panel DSNTIPX, or ZPARM STORMXAB in DSN6SYSP.

Field Name: QWP1SPAB

TIMEOUT VALUE

The number of seconds before DB2 stops waiting for an SQL CALL statement to be assigned to one of the TCBs in the DB2 stored procedures address space.

Install parameter TIMEOUT VALUE on panel DSNTIPX, or ZPARM STORTIME in DSN6SYSP.

Field Name: QWP1SPTO
**IFCID 106 - System Initialization Parameters**

This topic shows detailed information about “Record Trace - IFCID 106 - System Initialization Parameters”.

When this block contains names that are too long for the space available, they are truncated. The full name is shown in the list of long names, which is printed at the end of this block. When present, the list shows the parameter identifier, in alphabetic order, and the complete name. If the name is too long for one line, it continues on the next.

**Record trace - IFCID 106 - System Initialization Parameters**

The field labels shown in the following sample layout of “Record Trace - IFCID 106 - System Initialization Parameters” are described in the following section.

### SYSTEM INITIALIZATION PARAMETERS

- **CHECKPOINT FREQUENCY**: 500000
- **MONITOR BUFFER SIZE**: 1048576
- **SERVICE UNIT LIMIT**: 0
- **STATS INTERVAL**: 1
- **SYSTEM INITIALIZATION PARAMETERS**
  - **PSEUDO CLOSE CHECKPOINTS**: 5
- **LEVEL ID CHECKPOINTS**: 5
- **ROLL UP PARALLEL THREAD**: NO
- **USER LOB VALUE STORE**: 10240
- **EXTRA BLOCKS REQ**: 100
- **ONL DSE SS STAT INTERVAL**: 5
- **USER XML VALUES (KB)**: 204800
- **CHECKPOINT TYPE**: 32
- **CHECKPOINT MODULE**: DSNZPARM
- **ACCESS CONTROL**: DSN00XAC
- **IDENTIFY/AUTH**: DSN39ATH
- **SIGNON**: DSN99X

#### LIST OF LONG NAMES

- **RLIMIT TABLE AUTHID**

---

**CHECKPOINT FREQUENCY**

Checkpoint frequency. This shows either the number of minutes (1 through 60) or the number of DB2 log records between the start of successive checkpoints. DB2 starts a new checkpoint when this value is reached.

You can use the SET LOG command to change the number of log records between checkpoints dynamically. Valid values are 1-60 when specifying a time value and 200-16000000 when specifying a number of records.

Install parameter CHECKPOINT FREQ on panel DSNTIPL, ZPARM CHKFREQ in DSN6SYSYP.

**Field Name**: QWP1LOGL

**TRACE TABLE SIZE (4K)**

Shows the size of the RES trace table in 4 KB blocks. A value of 16 means 64 KB have been allocated for this table.

This is the default destination for the global trace records in DB2. Most trace records require 32-byte entries; events with more than three data items require 64-byte entries.

Install parameter TRACE SIZE on panel DSNTIPN, or ZPARM TRACTBL in DSN6SYSYP.

**Field Name**: QWP1TRSZ

**GLOBAL CLASSES**
**IFCID 106 - System Initialization Parameters**

Shows whether the global trace is started automatically when DB2 is started.

When YES, the global trace starts for the default classes (classes 1, 2, and 3) whenever DB2 is started, and additional data consistency checks are made whenever a data page or index page is modified. When ALL, the global trace is automatically started for all classes.

The global trace is used to diagnose problems in DB2 but it also impacts DB2 performance. If you have production systems requiring high performance, you might consider turning off global trace. If you do this, be aware that this presents a serviceability exposure. In the event of a system failure, IBM service personnel will ask you to turn on global trace and attempt to recreate the problem.

Install parameter TRACE AUTO START on panel DSNTIPN, or ZPARM TRACSTR in DSN6SYSP.

Field Name: QWP1TRST

**WTO ROUTE CODES**

The MVS console routing codes.

These codes are assigned to messages that are not solicited from a specific console. Up to 16 comma-separated codes can be shown.

Install parameter WTO ROUTE CODES on panel DSNTIPO, or ZPARM ROUTCDE in DSN6SYSP.

Field Name: QWP1SMRC

**MONITOR BUFFER SIZE**

The default number of bytes allocated for the monitor trace buffer.

Install parameter MONITOR SIZE on panel DSNTIPN, or ZPARM MONSIZE in DSN6SYSP.

Field Name: QWP1MONS

**BACKGROUND IDS**

The maximum allowed number of concurrent connections for batch jobs and utilities. This includes:

- All batch jobs using QMF.
- All batch jobs using the DSN command processor.
- All tasks connected to DB2 through call attach facility (CAF) running in batch. This can include:
  - Batch jobs using QMF
  - APPC applications
  - TCP/IP FTP connections

When the number of batch jobs reaches this limit, further requests are rejected.

Install parameter MAX BATCH CONNECT on panel DSNTIPE, or ZPARM IDBACK in DSN6SYSP.

Field Name: QWP1IDB

**STATISTICS CLASSES**
IFCID 106 - System Initialization Parameters

Shows whether a Statistics trace was started automatically at DB2 startup time.
The classes started are shown separated by commas.
DB2 sends collected trace data to SMF. The SMFPRM xx member of SYS1.PARMLIB must be set to allow SMF to write the records.
Install parameter SMF STATISTICS on panel DSNTIPN, or ZPARM SMFSTAT in DSN6SYSP.

Field Name: QWP1SMFS

RLIMIT TABLE ID

The default resource limit specification table (RLST) suffix.
This suffix is used when the resource limit facility (governor) is automatically started or when the governor is started without specifying a suffix.
Install parameter RLST NAME SUFFIX on panel DSNTIPO, or ZPARM RLFTBL in DSN6SYSP.

Field Name: QWP1RLFT

SERVICE UNIT LIMIT

The action taken by DB2 when the governor cannot use the resource limit specification table:

NOLIMIT
The dynamic SQL statements run without limit.

NORUN
The dynamic SQL statements are immediately terminated with an SQL error code.

A number from 1 to 5000000 represents the number of CPU service units allowed for a query.
Install parameter RLST ACCESS ERROR on panel DSNTIPO, or ZPARM RLFERR in DSN6SYSP.

Field Name: QWP1RLFN

FOREGROUND IDS

The maximum number of concurrent TSO foreground connections (QMF, DSN, DB2I, and SPUFI).
Each of the following is a separate user:
• Each TSO foreground user executing a DSN command.
• Each TSO foreground user connected to DB2 through the call attachment facility (CAF). This can include QMF users running in TSO foreground or user-written CAF applications running in TSO foreground.
When the number of TSO users attempting to access DB2 exceeds this limit, connection requests are rejected.
There is no subsystem parameter to control the maximum concurrent connections for IMS and CICS. These are controlled by using IMS and CICS facilities. For CICS attachment, the maximum number of connections to DB2 can be controlled using the resource control table (RCT) TYPE=INIT THRD MAX value.
IFCID 106 - System Initialization Parameters

Install parameter MAX TSO CONNECT on panel DSNTIPE, or ZPARM IDFORE in DSN6SYSP.

Field Name: QWP1IDF

ACCOUNTING CLASSES

Shows whether DB2 sends accounting data to SMF automatically when DB2 is started. Numeric values show what classes are sent. When YES, the default class (class 1) is sent. When ALL, accounting classes one through five are started.

The SMFPRM xx member of SYS1.PARMLIB must also be set to allow SMF to write the records.

Install parameter SMF ACCOUNTING on panel DSNTIPN, or ZPARM SMFACCT in DSN6SYSP.

Field Name: QWP1SMFA

RLIMIT FLAGS

The action taken by DB2 when the governor cannot use the resource limit specification table:

NOLIMIT
   The dynamic SQL statements run without limit.

NORUN
   The dynamic SQL statements are immediately terminated with an SQL error code.

A number from 1 to 5000000 represents the number of CPU service units allowed for a query.

Install parameter RLST ACCESS ERROR on panel DSNTIPO, or ZPARM RLFERR in DSN6SYSP.

Field Name: QWP1RLFR

STATS INTERVAL

The time interval, in minutes, between statistics collections. Statistics records are written approximately at the end of this interval.

Install parameter STATISTICS TIME on panel DSNTIPN, or ZPARM STATIME in DSN6SYSP.

Field Name: QWP1STIM

CONCURRENT THREADS

The maximum number of allied threads (threads started at the local subsystem) that can be allocated concurrently.

Separate threads are created for each occurrence of the following:
- TSO user (whether running a DSN command or a DB2 request from QMF)
- Batch job (whether running a DSN command or a DB2 utility)
- IMS region that can access DB2
- Active CICS transaction that can access DB2
- Task connected to DB2 through the call attachment facility.

Install parameter MAX USERS on panel DSNTIPE, or ZPARM CTHREAD in DSN6SYSP.
IFCID 106 - System Initialization Parameters

Field Name: QWP1CT

AUDIT CLASSES

Shows whether the audit trace is started automatically when DB2 is started.

When YES, the audit trace is started for the default class (class 1) whenever DB2 is started. When ALL, an audit trace is automatically started for all classes.

Install parameter AUDIT TRACE on panel DSNTIPN, or ZPARM AUDITST in DSN6SYSP.

Field Name: QWP1AUDT

EXT. SECURITY

Extended security options.

When YES (strongly recommended), detailed reason codes are returned to a DRDA level 3 client when a DDF connection request fails because of security errors. When using SNA protocols, the requester must have included a product that supports the extended security sense codes, such as DB2 Connect™ version 5 and subsequent releases.

RACF users can change their passwords using the DRDA change password function. This support is only for DRDA level 3 requesters that have implemented support for changing passwords.

YES allows properly enabled DRDA clients to determine the cause of security failures without requiring DB2 operator support.

When NO, generic error codes are returned to the clients and RACF users are prevented from changing their passwords.

Install parameter EXTENDED SECURITY on panel DSNTIPR, or ZPARM EXTSEC in DSN6SYSP.

Field Name: QWP1SCER

PSEUDOCLOSE CHECKPOINTS

The number of consecutive DB2 checkpoints that a page set or partition can remain in read/write mode since it was last updated. When this limit or the RO SWITCH TIME is reached, DB2 changes the page set or partition to read only.

This can improve performance for recovery, logging, and data-sharing processing.

Install parameter RO SWITCH CHKPTS on panel DSNTIPL, or ZPARM PCLOSEN in DSN6SYSP.

Field Name: QWP1FREQ

REMOTE THREADS(ACTIVE)

The maximum number of database access threads (DBATs) that can be active concurrently.

When this limit has been reached, DB2 uses the value of DDF THREADS on panel DSNTIPR to decide how to handle a new allocation request.
When DDF THREADS is ACTIVE and MAX REMOTE CONNECTED has not been reached, the allocation request is allowed but any further processing for the connection is queued waiting for an active database access thread to terminate.

When DDF THREADS is INACTIVE and MAX REMOTE CONNECTED has not been reached, the allocation request is allowed and is processed when DB2 can assign an unused database access thread slot to the connection.

The total number of threads accessing data concurrently is the sum of MAX USERS and MAX REMOTE ACTIVE. The maximum allowable value for this sum is 2000.

Install parameter MAX REMOTE ACTIVE on panel DSNTIPE, or ZPARM MAXDBAT in DSN6SYSP.

Field Name: QWP1RMT

**MONITOR CLASSES**

Shows whether the monitor trace is started automatically when DB2 is started. When YES, the default (trace class 1) is started. Numeric values show which classes are started. When ALL, monitor trace classes 1 through 8 are started.

Install parameter MONITOR TRACE on panel DSNTIPN, or ZPARM MON in DSN6SYSP.

Field Name: QWP1MON

**LIMIT BACKOUT**

Shows whether some backward log processing should be postponed. When NO, DB2 backward log processing processes all inflight units of recovery (URs) and URs for abending transactions. When YES, DB2 postpones backout processing for some units of work until the command RECOVER POSTPONED is issued. AUTO (default) postpones some backout processing but automatically starts the backout processing when DB2 restarts and begins accepting new work.

When YES or AUTO, backout processing runs concurrently with new work. Page sets or partitions with backout work pending are unavailable until their backout work is complete.

Install parameter LIMIT BACKOUT on panel DSNTIPL, or ZPARM LBACKOUT in DSN6SYSP.

Field Name: QWP1LMBO

**PSEUDOCLOSE MINUTES**

The number of minutes that a page set or partition can remain in read-write mode since it was last updated. When this limit or the RO SWITCH CHKPTS is reached, DB2 changes the page set or partition to read-only.

This can improve performance for recovery, logging, and data-sharing processing.

Install parameter RO SWITCH TIME on panel DSNTIPL, or ZPARM PCLOSET in DSN6SYSP.
IFCID 106 - System Initialization Parameters

Field Name: QWPITMR
REMOTE THREADS(CONNECT)

The maximum allowed number of concurrent remote connections.
When this limit is reached, any new connection request is rejected.
Install parameter MAX REMOTE CONNECTED on panel DSNTIPE, or ZPARM CONDBAT in DSN6SYSP.

Field Name: QWPICDB
RLIMIT TABLE AUTHID

The authorization ID used for the resource limit facility (governor).
This identifier can be a long string. If there is insufficient space to show the complete string, the string is truncated in the report block. The complete string is shown in a separate list of long names at the end of the report.
Install parameter RESOURCE AUTHID on panel DSNTIPP, or ZPARM RLFAUTH in DSN6SYSP.

Field Name: QWP1RLF
BACKOUT DURATION

Indicates how much of the log to process for backout when LIMIT BACKOUT = YES or AUTO.
During restart, backward log processing continues until both of the following events occur:
• All inflight and inabort URs with update activity against the catalog or directory are backed out.
• The number of log records processed is equal to the number specified in BACKOUT DURATION times the value of CHECKPOINT FREQ. If the checkpoint frequency is specified in minutes, the number of records processed is the default of 50000 records multiplied by the value of CHECKPOINT FREQ.
In-flight and in-abort URs that are not completely backed out during restart are converted to postponed-abort status. Page sets or partitions with postponed-backout work are put into restart pending (RESTP). This state blocks all access to the object other than access by the command RECOVER POSTPONED or by automatic backout processing performed by DB2 when LIMITED BACKOUT = AUTO.
A table space might be in restart pending mode, without the associated index spaces also in restart pending mode. This happens if a postponed abort UR makes updates only to non-indexed fields of a table in a table space. In this case, the indexes are accessible to SQL (for index-only queries), even though the table space is inaccessible.
Install parameter BACKOUT DURATION on panel DSNTIPL, or ZPARM BACKODUR in DSN6SYSP.

Field Name: QWP1BDUR
LEVEL ID CHECKPOINTS

How often, in checkpoints, the level ID of a page set or partition is updated. When zero (0), downlevel detection is disabled.
Use the following criteria to decide on a suitable value for this parameter:
IFCID 106 - System Initialization Parameters

- **How often are backup and restore methods used outside of the DB2 control (such as DSN1COPY or DFDSS dump and restore)?** If rarely used, there is no need to update the level ID frequently.

- **How many page sets are open for update at the same time?** If DB2 updates level IDs frequently, there is extra protection against downlevel page sets. However, a performance degradation can occur if the level IDs for many page sets must be set at every checkpoint.

- **How often does the subsystem take checkpoints?** If the DB2 subsystem takes frequent system checkpoints, set the level ID frequency to a higher value.

Install parameter LEVELID UPDATE FREQ on panel DSNTIPL, or ZPARM DLDFREQ in DSN6SYSP.

**Field Name:** QWP1DFRFQ

**UR CHECK FREQUENCY**

Shows the number of checkpoint cycles to complete before DB2 issues a warning message to the console and writes an IFCID 313 record for an uncommitted, indoubt, or inflight unit of recovery (UR). The default is 0, which disables this option.

Install parameter UR CHECK FREQ on panel DSNTIPL, or ZPARM URCHKTH in DSN6SYSP.

**Field Name:** QWP1URCK

**WLM ENVIRONMENT**

Workload manager environment.

Install parameter WLM ENVIRONMENT on panel DSNTIPX, or ZPARM WLMENV in DSN6SYSP.

**Field Name:** QWP1WLME

**DATABASE PROTOCOL**

The default protocol (DRDA or PRIVATE) used when option DBPROTOCOL BIND is not explicitly specified for the bind of a plan or a package.

When field INSTALL TYPE on panel DSNTIPA1 is INSTALL, the default value for DATABASE PROTOCOL is DRDA. When the value of INSTALL TYPE is MIGRATE, the default value for DATABASE PROTOCOL is PRIVATE.

An application program might contain statements with three-part names, or aliases that reference remote objects. At bind or rebind of a plan, a user can specify whether these statements flow to the remote site using DB2 private or DRDA protocol.

DB2 private protocol is appropriate if you do not plan to move applications that use three-part names to DRDA access immediately. To use DRDA access for applications with three-part names, you must bind packages for those applications at each location that the applications access, then bind all packages into a plan. If you cannot perform this activity immediately, and you want your applications to continue to work, you should specify PRIVATE for DATABASE PROTOCOL.

The BIND commands for DB2-supplied applications are in job DSNTIJSG.
IFCID 106 - System Initialization Parameters

Install parameter DATABASE PROTOCOL on panel DSNTIP5, or ZPARM DBPROTCL in DSN6SYSP.

**Field Name:** QWP1DBPR

**ROLL UP PARALLEL THREAD**

Indicates whether DB2 generates a trace record at the originating task level that summarizes accounting information for all parallel tasks.

DB2 parameter PTASKROL in DSN6SYSP.

**Field Name:** QWP1PROL

**LOCAL TRACE TAB SIZE**

The size of the local trace tables in multiples of 4 KB.

**Field Name:** QWP1TLSZ

**DEF 4K BP USER DATA**

The name of the 4 KB buffer pool for user table spaces.

Install parameter DEFAULT BUFFER POOL FOR USER DATA on installation panel DSNTIP1, or ZPARM TBSBPOOL in DSN6SYSP.

**Field Name:** QWP1TBPL

**USER LOB VALUE STOR IN KB**

The maximum amount of storage (KB) each user can use for LOB values.

Install parameter USER LOB VALUE STORAGE on panel DSNTIP7, or ZPARM LOBVALA in DSN6SYSP.

**Field Name:** QWP1LVA

**SYS LOB VALUE STOR IN MB**

The maximum amount of storage (MB) each system can use for LOB values.

Install parameter SYSTEM LOB VALUE STORAGE on panel DSNTIP7, or ZPARM LOBVALS in DSN6SYSP.

**Field Name:** QWP1LV

**DEF BPOOL USER INDEX**

The name of the 4 KB buffer pool used for indexes on user data.

Install parameter DEFAULT BUFFER POOL FOR USER INDEXES on installation panel DSNTIP1, or ZPARM IDXBPOOL in DSN6SYSP.

**Field Name:** QWP1IXPL

**DBM1 ST FAST LOG**

The maximum DBM1 storage that can be used by the fast log apply process. The default value is 0 MB, which means that fast log apply is disabled except during DB2 restart, when fast log apply is always enabled.

Install parameter LOG APPLY STORAGE on panel DSNTIPL, or ZPARM LOGAPSTG in DSN6SYSP.

**Field Name:** QWP1FLBZ

**EXTRA BLOCKS REQ**
IFCID 106 - System Initialization Parameters

The maximum number of extra DRDA query blocks DB2 requests from a remote DRDA server.

The default is 100.

This controls the total amount of data that can be transmitted on any given network exchange. It does not limit the size of the SQL query answer set.

Install parameter EXTRA BLOCKS REQ on panel DSNTIP5, ZPARM EXTRAREQ in DSN6SYSP.

Field Name: QWP1EXBR

EXTRA BLOCKS SRV

The maximum number of extra DRDA query blocks DB2 returns to a DRDA client.

The default is 100.

This controls the total amount of data that can be transmitted on any given network exchange. It does not limit the size of the SQL query answer set.

Install parameter EXTRA BLOCKS SRV on panel DSNTIP5, ZPARM EXTRASRV in DSN6SYSP.

Field Name: QWP1EXBS

INTERVAL SYNCHR W/IN HOUR

Shows whether DB2 statistics recording is synchronized with some part of the hour. The installation can specify that the DB2 statistics recording interval be synchronized with the beginning of the hour (00 minutes past the hour) or any number of minutes past the hour up to 59. Possible values are: 0-59, which indicate the synchronization point. When NO or N/A is shown, synchronization is disabled, this is the default.

If STATISTICS TIME INTERVAL IN MINUTES (STATIME) is greater than 60, NO or N/A is shown.

Install parameter STATISTICS SYNC on panel DSNTIPN, or ZPARM SYNCVAL in DSN6SYSP.

Field Name: QWP1SYNV

SYNCHR FLAG

Interval synchronization flag, shows whether the synchronization is enabled. The default is NO. When YES, the DB2 statistics are synchronized to the value shown in INTERVAL SYNCHR W/IN HOUR.

Field Name: QWP1SYFL

ONL DSET STATS INTERVAL

The time interval, in minutes, before DB2 resets data set statistics collected for the online performance monitors. Online performance monitors can request DB2 data set statistics for the current interval with an IFI READS request for IFCID 199.

Install parameter DATASET STATS TIME on panel DSNTIPN, or ZPARM DSSTIME in DSN6SYSP.

Field Name: QWP1DTIM

DDF/RRSAF ACCUM
**IFCID 106 - System Initialization Parameters**

Shows whether DB2 accounting data for DDF and RRSAF threads is accumulated by end user.

When NO, DB2 writes an accounting record when a DDF thread is made inactive, or when signon occurs for an RRSAF thread. A value in the range 2 through 65535 shows the number of times an end-user identifier should occur before DB2 writes an accounting record. An end-user identifier is the concatenation of the end-user user ID, end-user transaction name, and the end-user workstation name.

These values can be set by DDF threads using SERVER CONNECT and SET CLIENT calls, and by RRSAF threads using the RRSAF SIGN, AUTH SIGNON, and CONTEXT SIGNON functions.

An accounting record might be written prior to the number of end user occurrences in the following instances:

- When an internal storage threshold is reached for the accounting RRSAF signon call.
- When the thread deallocates, the accumulated accounting data for all end users on this thread is written (one record per end user).
- When this parameter is dynamically changed to deactivate accounting accumulation. In this instance, the next end-UR (for DDF thread) or signon (for a RRSAF thread) causes DB2 to write the accumulated accounting data for all end users on this thread (one record per end user).

Install parameter DDF/RRSAF ACCUM on installation panel DSNTIPN, or ZPARM ACCUMACC in DSN6SYSP.

**Field Name:** QWP1ACCU

**TS ALLOCATION**

Shows the amount of space in KB for primary and secondary space allocation for DB2-defined data sets for table spaces created without the USING clause. 0 indicates that DB2 uses standard defaults.

Install parameter TABLE SPACE ALLOCATION on panel DSNTIP7, or ZPARM TSQTY in DSN6SYSP.

**Field Name:** QWP1TSQT

**IX ALLOCATION**

Shows the amount of space in KB for primary and secondary space allocation for DB2-defined data sets for index spaces created without the USING clause. 0 indicates that DB2 uses standard defaults.

Install parameter INDEX SPACE ALLOCATION on panel DSNTIP7, or ZPARM IXQTY in DSN6SYSP.

**Field Name:** QWP1IXQT

**UR LOG THRESHOLD**

Shows the number of log records that are to be written by an uncommitted unit of recovery (UR) before DB2 issues a warning message to the console. This provides notification of a long-running UR. Long-running URs might result in a lengthy DB2 restart or a lengthy recovery situation for critical tables. Log records are specified in 1-K (1000 log records) increments. A value of 0 indicates that no write check is to be performed.
**IFCID 106 - System Initialization Parameters**

Install parameter UR LOG WRITE CHECK on panel DSNTIPL, ZPARM URLGHT in DSN6SYS.

**Field Name:** QWP1LWCK

**UNICODE IFCIDS**

Shows whether output from IFC records should include Unicode information. Only a subset of the character fields (identified in the IFCID record definition by a %U in the comment area to the right of the field declaration in the DSNDQWxx copy files) are encoded in Unicode. The remaining fields maintain the same encoding of previous releases.

Install parameter UNICODE IFCIDS on panel DSNTIPN, or ZPARM UIFICIDS in DSN6SYS.

**Field Name:** QWP1_UNICODE

**AGGREGATION FIELDS**

Shows the aggregation fields used for DDF and RRSAF accounting rollup. Values are defined as follows:

0   End user ID, transaction name, and workstation name
1   End user ID
2   End user transaction name
3   End user workstation name
4   End user ID and transaction name
5   End user ID and workstation name
6   End user transaction name and workstation name

This value is ignored if DDF or RRSAF accounting are not used. DB2 writes individual accounting threads for threads that do not have all aggregation fields populated that are specified by this parameter.

Install parameter AGGREGATION FIELDS on installation panel DSNTIPN, or ZPARM ACCUMUID in DSN6SYS.

**Field Name:** QWP1ACID

**VARY DS CONTR INTVAL**

Indicates whether DB2 optimizes VSAM CONTROL INTERVAL to page size for data set allocation.

Install parameter VARY DS CONTROL INTERVAL on panel DSNTIP7, or ZPARM DSVCI in DSN6SYS.

**Field Name:** QWP1VVCI

**OPTIMIZE EXTENT SIZING**

Indicates whether DB2 uses sliding secondary quantity for DB2 managed data sets to optimize extent sizing.

Install parameter OPTIMIZE EXTENT SIZING on panel DSNTIP7, or ZPARM MGEXTSZ in DSN6SYS.

**Field Name:** QWP1MESZ

**DEFINE DATA SETS**
IFCID 106 - System Initialization Parameters

Defines the underlying data sets when a table space (TS) that is contained in an implicitly created database is created.

Install parameter DEFINE DATA SETS on panel DSNTIP7 or ZPARM IMPDSDEF in DSN6SYSP.

**Field Name:** QWP1DIDS

**USE DATA COMPRESSION**

Shows whether data compression in table spaces in implicitly defined databases is used.

Install parameter USE DATA COMPRESSION on panel DSNTIP7 or ZPARM IMPTSCMP in DSN6SYSP.

**Field Name:** QWP1CITS

**DEL CF STRUCTS**

Shows whether to:

- Delete change-data (CD) structures during restart
- Attempt to delete coupling-facility (CF) structures, including shared communications area (SCA) structures, internal resource lock manager (IRLM lock) structures, and allocated group buffer pools.

This field corresponds to field DEL CF STRUCTS on installation panel DSNTIPK.

ZPARM name DEL_CFSTRUCTS_ON_RESTART in DSN6SYSP.

**Field Name:** QWP1DCFS

**MAX OPEN DS FOR LOB**

The maximum number of concurrently open data sets for processing LOB file references.

Install parameter MAX OPEN FILE REFS on panel DSNTIPE or ZPARM MAXOFILR in DSN6SYSP.

**Field Name:** QWP1MOFR

**LOB INLINE LENGTH**

Default inline length for any new storing large object (LOB) column in a Universal Table Space on the DB2 subsystem. The valid values are from 0 to 32680 inclusive (in bytes). The default value for this ZPARM is 0, which indicates that no inline attribute is desired for any LOB column (BLOB, CLOB or DBCLOB) created on this subsystem.

**Field Name:** QWP1LBIL

**COMPRESS SMF RECS**

Shows the COMPRESS DEST(SMF) TRACE records. This field corresponds to field COMPRESS SMF RECS on installation panel DSNTIPN. ZPARM name: SMFCOMP in DSN6SYSP.

**Field Name:** QWP1CSMF

**RANDOMIZE XML DOCID**

Specifies whether DB2 generates document ID elements sequentially or randomly. Possible values are:

- **YES**  Sequentially
- **NO**   Randomly
NO Randomly
ZPARM name XML_RANDOMIZE_DOCID in DSN6SYSP.

Field Name: QWP1XRDI

DEF 8K BP USER DATA
The default 8 KB buffer pool for:
- Table spaces with an 8 KB page size in implicitly created databases
- Explicitly created table spaces with an 8 KB page size, but without a buffer pool clause that is specified in the create table space statement.

Install parameter DEFAULT 8-KB BUFFER POOL FOR USER DATA on panel DSNTP1 or ZPARM TBSBP8K in DSN6SYSP.

Field Name: QWP1TP8

DEF 16K BP USER DATA
The default 16 KB buffer pool for:
- Table spaces with a 16 KB page size in implicitly created databases
- Explicitly created table spaces with a 16 KB page size, but without a buffer pool clause that is specified in the create table space statement.

Install parameter DEFAULT 16-KB BUFFER POOL FOR USER DATA on panel DSNTP1 or ZPARM TBSBP16K in DSN6SYSP.

Field Name: QWP1TP16

DEF 32K BP USER DATA
The default 32 KB buffer pool for:
- Table spaces with a 32 KB page size in implicitly created databases
- Explicitly created table spaces with a 32 KB page size, but without a buffer pool clause that is specified in the create table space statement.

Install parameter DEFAULT 32-KB BUFFER POOL FOR USER DATA on panel DSNTP1 or ZPARM TBSBP32K in DSN6SYSP.

Field Name: QWP1TP32

USER XML VALUES (KB)
The maximum amount of memory in kilobytes (KB) for each user for storing XML values.
ZPARM XMLVALA in DSN6SYSP.

Field Name: QWP1XVA

SYSTEM XML VALUES (MB)
The maximum amount of memory in megabytes (MB) for each system for storing XML values.
ZPARM XMLVALS in DSN6SYSP.

Field Name: QWP1XVS

DEF PART SEGSIZE
The default segment size to be used for a partitioned table space when the CREATE TABLESPACE statement does not include the SEGSIZE parameter.
This field corresponds to field DEFAULT PARTITION SEGSIZE on installation panel DSNTP7. The ZPARM name is DPSEGSZ in DSN6SYSP.
IFCID 106 - System Initialization Parameters

**Field Name:** QWP1DPSS

**USE TRACKMOD (IMPLICIT TS)**

Shows whether you have specified the TRACKMOD option on ALTER TABLESPACE for an implicitly created table space.

This field corresponds to field TRACK MODIFIED PAGES on installation panel DSNTIP7. The ZPARM name is IMPTKMOD in DSN6SPRM.

**Field Name:** QWP1TKMD

**DSSIZE (IMPLICIT TS)**

Shows the maximum DSSIZE in gigabytes that DB2 uses for creating each partition of an implicitly created base table space.

This field corresponds to field DEFAULT DSSIZE on installation panel DSNTIP7. The ZPARM name is IMPDSSIZE in DSN6SPRM.

**Field Name:** QWP1DSSZ

**CHECKPOINT TYPE**

Shows the LOG checkpoint type. It can have the following values:

- **SINGLE**
  - Either records or minutes.

- **BOTH**
  - Both records and minutes, as specified by Records Between Checkpoint (QWP1LOGR) and Mins Between Checkpoint (QWP1LOGM).

ZPARM CHKTYPE in DSN6SYSP.

**Field Name:** QWP1LOGT

**RECORDS/CHECKPOINT**

Shows the number of records between log checkpoints if the LOG checkpoint type is BOTH (records and minutes).

This field corresponds to field RECORDS/CHECKPOINT on installation panel DSNTIPL1, or ZPARM name CHKLOGR in DSN6SYSP.

**Field Name:** QWP1LOGR

**MINUTES/CHECKPOINT**

Shows the number of minutes between log checkpoints if the LOG checkpoint type is BOTH (records and minutes).

This field corresponds to field MINUTES/CHECKPOINT on installation panel DSNTIPL1, or ZPARM name CHKMINS in DSN6SYSP.

**Field Name:** QWP1LOGM

**PARAMETER MODULE**

Shows the name of the active subsystem parameter module.

This field corresponds to field PARAMETER MODULE on installation panel DSNTIPO3.

**Field Name:** QWP1ZPNM

**ACCESS CONTROL**

Shows the name of the default access control exit module.
IFCID 106 - System Initialization Parameters

This field corresponds to field ACCESS CONTROL on installation panel DSNTIPO3. The ZPARM name is ACCESS_CNTL_MODULE in DSN6SYSP.

Field Name: QWP1DXAC

IDENTIFY/AUTH

Shows the name of the default identify or authorization exit module.

This field corresponds to field IDENTIFY/AUTH on installation panel DSNTIPO3. The ZPARM name is IDAUTH_MODULE in DSN6SYSP.

Field Name: QWP1DATH

SIGNON

Shows the name of the default signon exit module.

This field corresponds to field SIGNON on installation panel DSNTIPO3. The ZPARM name is SIGNON_MODULE in DSN6SYSP.

Field Name: QWP1DSGN

QWP1DB1M

The amount of space reserved for MVS functions.

Field Name: QWP1DB1M

QWP1CRIT

The amount of space reserved for critical work that must be completed.

Field Name: QWP1CRIT

QWP1SOS

The amount of space above MVS and critical (QWP1DB1M + QWP1CRIT) DB2 tries to leave available.

Field Name: QWP1SOS

QWP1LVL

Level of this block. It is used to detect parameters or code that is out of sync.

Field Name: QWP1LVL

QWP1FLAG

The following flags indicate:

X'80' Indicates whether any allies can be swapped. (1=no allies swappable)

X'20' Indicates whether the Accounting trace of a parallel task of a query should be rolled into the originating Accounting trace of the task. (1=roll up accounting traces)

X'10' Indicates whether fields that contain %U in their comments are in Unicode (UTF-8). (1=trace in Unicode.)

X'08' Indicates whether to use CI SIZE=PAGE size when defining DB2-managed data sets. (1=use CI SIZE=page size.)

X'04' Enable a sliding scale for secondary space allocations for DB2-managed data sets.

X'02' Detailed measured usage price tracking.

Field Name: QWP1FLAG
IFCID 106 - System Initialization Parameters

RLIMIT TABLE AUTHID

The authorization ID used for the resource limit facility (governor).

This identifier can be a long string. If there is insufficient space to show the complete string, the string is truncated in the report block. The complete string is shown in a separate list of long names at the end of the report.

Install parameter RESOURCE AUTHID on panel DSNTIPP, or ZPARM RLFAUTH in DSN6SYSP.

Field Name: QWP1RLFA
IFCID 106 - VSAM Catalog Name Qualifier

This topic shows detailed information about “Record Trace - IFCID 106 - VSAM Catalog Name Qualifier”.

Record trace - IFCID 106 - VSAM Catalog Name Qualifier

The field labels shown in the following sample layout of “Record Trace - IFCID 106 - VSAM Catalog Name Qualifier” are described in the following section.

VSAM CATALOG NAME QUALIFIER

DSN851

Qualifiers

The high-order qualifier name of all DB2 system data sets.

Field Name: QWP6CATN
IFCID 107 - Open/Close

This topic shows detailed information about “Record Trace - IFCID 107 - Open/Close”.

Record trace - IFCID 107 - Open/Close

The field labels shown in the following sample layout of “Record Trace - IFCID 107 - Open/Close” are described in the following section.

DBID: 5  DATABASE NAME: DSNDB07
OBID: 24  OBJECT NAME: DSNDSX02

DBID

The decimal identifier of the database.

Field Name: QW0107DB

DATABASE NAME

The name of the database.

Field Name: QW0107DN

OBID

The decimal identifier of the object. Examples of objects are table space and index space.

Field Name: QW0107OB

OBJECT NAME

The name of the object. Examples of objects are table space and index space.

Field Name: QW0107TN
IFCID 108 - Bind Start

This topic shows detailed information about “Record Trace - IFCID 108 - Bind Start”.

Record trace - IFCID 108 - Bind Start

The field labels shown in the following sample layout of “Record Trace - IFCID 108 - Bind Start” are described in the following section.

LOCATION

The package location.

Field Name: QW0108NL

SQLERROR

Indicates whether a package is created if SQL errors are encountered:

CONTIN A package is created even if SQL errors are encountered.

NOPACK A package is not created if SQL errors are encountered.

Field Name: QW0108E

REBIND PLAN(*)

Indicates whether a plan is rebound.

Field Name: QW0108S

ISOLATION

The isolation level for plans and packages:

RR Repeateable read

CS Cursor stability

RS Read stability

UR Uncommitted read

For packages only:

DF Default (at run time, assumes the isolation level of the current plan)

Field Name: QW0108I

RELEASE

Indicates when to release the locks:

COMMIT Release locks at commit time.

DEALLOC Release locks at deallcation time.
IFCID 108 - Bind Start

For packages only:

**DEFAULT**
Release locks at run time, which is the default.

**Field Name:** QW0108R

**COLLECT ID**
The collection identifier of the package.

**Field Name:** QW0108NC

**DEGREE**
The degree bind option:

**ANY** Degree(any)

**1** Degree(1)

**Field Name:** QW0108PL

**EXPLAIN**
Indicates whether EXPLAIN was specified.

**Field Name:** QW0108X

**TYPE**
The type of bind.

**Field Name:** QW0108T

**QUALIFIER**
The qualifier used for unqualified object names.

**Field Name:** QW0108QL

**PACKAGE ID**
The package identifier.

**Field Name:** QW0108NI

**SQLRULES**
The SQL rules option.

**Field Name:** QW0108SR

**OWNER**
The plan or package owner.

**Field Name:** QW0108OW

**ACTION**
Indicates whether the plan or package replaces an existing plan or package with the same name or is new. This field only applies to BIND activities.

**Field Name:** QW0108A

**CACHE SIZE**
The authorization cache size. A value of 0 indicates that DB2 determines the size.

**Field Name:** QW0108CA
TOKEN
The consistency token of the package.
Field Name: QW0108NT

DISCONNECT
The disconnect option:
- **EXPL**  Explicit
- **AUTO**  Automatic
- **COND**  Conditional
Field Name: QW0108DC

OBJECT TYPE
The type of object bound or rebound.
Field Name: QW0108TY

VALIDATION
The time of validation:
- **RUN**  Validate at run time.
- **BIND**  Validate at bind time.
Field Name: QW0108V

REOPT
Indicates whether reoptimization was requested:
- **YES**  REOPT(VARS) was specified to reoptimize the access path of the SQL statement at run time.
- **NO**  NOREOPT(VARS) was specified to optimize the access path of the SQL statement only at bind time.
Field Name: QW0108RO

PLAN NAME
The plan name. 'BLANK' indicates that a test bind was performed.
Field Name: QW0108PN

DYNAMICRULES
The value of the DYNAMICRULES option on the BIND/REBIND command:
- **RUN**  Run-time rules apply to a dynamic SQL statement for authorization checking and object qualification at run time.
- **BIND**  Bind-time rules apply to a dynamic SQL statement for authorization checking and object qualification at run time.
- **N/P**  DYNAMICRULES was not specified.
Field Name: QW0108DY

CURRENTDATA
Controls the data currency for ambiguous cursors:
Data currency is not required for ambiguous cursors. Blocking for ambiguous cursors is allowed.

YES  Data currency is required for ambiguous cursors. Blocking for ambiguous cursors is inhibited.

ALL  Data currency is required for all cursors. Applicable to packages only.

Field Name: QW0108F

ACQUIRE

Indicates when to acquire the locks:

ALLOC  Acquire the locks when the plan is allocated.

USE  Acquire the locks when the application first accesses them.

Field Name: QW0108Q

KEEPDYNAMIC

Indicates whether DB2 keeps (KEEPDYNAMIC(YES)) or discards (KEEPDYNAMIC(NO)) prepared SQL statements at commit points.

Field Name: QW0108KD

DBPROTOCOL

Database protocol option. Possible values are:

DRDA  DRDA protocol

PRIVATE  DB2 private protocol

Field Name: QW0108PR

DEFERPREPARE

Indicates whether preparation of dynamic SQL statements was deferred. Possible values are:

YES  Dynamic SQL statement preparation was deferred.

NO  Dynamic SQL statements were prepared immediately.

Field Name: QW0108DP

OPT_HINT_IDENT

Query optimization hint identifier, the default is blanks.

Field Name: QW0108OH

IMMEDWRITE

Indicates how DB2 updates group buffer pool dependent pages. This is only valid in a data-sharing environment.

Group buffer pool dependent pages can be written out to DASD or SYSTEM pagesets. Values shown are:

NO  DB2 uses normal write activity for updates, this is the default. Pages are written out at, or before phase 2 commit, or at the end of an abort for transactions that have rolled back.

PH1  Pages are written out at, or before phase 1 commit.
If a transaction subsequently rolls back, the pages are updated in the group buffer pool at the end of the rollback, and are written out at the end of the abort.

**YES** Pages are written out to the coupling facility as soon as the buffer update commits. Pages are written out regardless of whether the update occurs during forward progress or rollback of the transaction.

This option can affect DB2 performance due to coupling facility overhead.

**N/P** The DB2 subsystem is not part of a data sharing group.

**Field Name:** QW0108WI

**VERSION**

The version.

**Field Name:** QW0108VN
This topic shows detailed information about “Record Trace - IFCID 109 - Bind End”.

**Record trace - IFCID 109 - Bind End**

The field labels shown in the following sample layout of “Record Trace - IFCID 109 - Bind End” are described in the following section.

RETURN 0

**RETURN**

The bind return code:

- 0 Successful bind/rebind
- 4 Warning
- 8 Error

**Field Name:** QW0109RC
IFCID 110 - Bind Free Start

This topic shows detailed information about “Record Trace - IFCID 110 - Bind Free Start”.

Record trace - IFCID 110 - Bind Free Start

The field labels shown in the following sample layout of “Record Trace - IFCID 110 - Bind Free Start” are described in the following section.

BIND FREE --> 'BLANK'
START NETWORKID: DEIBMIPS LUNAME: IPSAU85C LUWSEQ: 1
PLAN : N/P FREE PLAN(*): N
OBJTYPE : PACKAGE TOKEN X'171BC6E10959E230'
LOCATION: PM05D85C
COLL-ID : DB2PM
PKG-ID : DGO8BCH
VERSION : R110_LEVE

PLAN
The name of the plan used in a bind.

Field Name: QW0110PN

FREE PLAN(*)
Indicates whether the command FREE PLAN(*) or FREE PACKAGE(*) was entered:
Y FREE PLAN(*) was entered.
N FREE PACKAGE(*) was entered.

Field Name: QW0110S

OBJTYPE
The type of object bound or rebound.

Field Name: QW0110TY

LOCATION
The package location.

Field Name: QW0110PL

COLL-ID
The collection identifier of the package.

Field Name: QW0110PC

PKG-ID
The package identifier.

Field Name: QW0110PI

TOKEN
The consistency token of the package.

Field Name: QW0110PT

VERSION
The version.

Field Name: QW0110VN
IFCID 111 - Bind Free End

Record trace - IFCID 111 - Bind Free End

The field labels shown in the following sample layout of “Record Trace - IFCID 111 - Bind Free End” are described in the following section.

RETURN

RETURN

The bind return code:

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Successful free plan</td>
</tr>
<tr>
<td>4</td>
<td>Warning</td>
</tr>
<tr>
<td>8</td>
<td>Error</td>
</tr>
</tbody>
</table>

Field Name: QW0111RC
IFCID 112 - Thread Allocate

This topic shows detailed information about “Record Trace - IFCID 112 - Thread Allocate”.

Record trace - IFCID 112 - Thread Allocate

The field labels shown in the following sample layout of “Record Trace - IFCID 112 - Thread Allocate” are described in the following section.

<table>
<thead>
<tr>
<th>FIELD</th>
<th>VALUE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>THREAD</td>
<td>--&gt; NETWORKID: DEIBMIPS LUNAME: IPSAU851 LUWSEQ: 1</td>
<td></td>
</tr>
<tr>
<td>ALLOCATE</td>
<td>PLAN NAME : DSNBIND ACQUIRE: ALLOCATION</td>
<td></td>
</tr>
<tr>
<td>ISOLATION</td>
<td>: CS RELEASE: DEALLOCATION</td>
<td></td>
</tr>
<tr>
<td>DYNAMICRULES</td>
<td>: RUN REDPT : NO</td>
<td></td>
</tr>
<tr>
<td>KEEPDYNAMIC</td>
<td>: NO PREPARE: NO</td>
<td></td>
</tr>
<tr>
<td>DBPROTOCOL</td>
<td>: PRIVATE</td>
<td></td>
</tr>
<tr>
<td>HINTID</td>
<td>: 'BLANK'</td>
<td></td>
</tr>
<tr>
<td>IMMEDWRITE</td>
<td>: NO</td>
<td></td>
</tr>
</tbody>
</table>

**PLAN NAME**

The plan name for the thread.

*Field Name: QW0112PN*

**ACQUIRE**

Indicates when to acquire locks:

**ALLOCATION**

Acquire the locks when the plan is allocated.

**USE**

Acquire the locks when the application first accesses them.

*Field Name: QW0112Q*

**ISOLATION**

The isolation level:

**RR** Repeatable read

**CS** Cursor stability

**RS** Read stability

**UR** Uncommitted read

*Field Name: QW0112I*

**RELEASE**

Indicates when to release locks:

**COMMIT** Release locks at commit time.

**DEALLOCATION** Release locks at deallocation time.

*Field Name: QW0112R*

**DYNAMICRULES**

The value of the DYNAMICRULES option on the BIND/REBIND command:

**RUN** Run-time rules apply to a dynamic SQL statement for authorization checking and object qualification at run time.
BIND  Bind-time rules apply to a dynamic SQL statement for authorization checking and object qualification at run time.

N/P  DYNAMICRULES was not specified.

Field Name: QW0112DY

REOPT  Indicates whether reoptimization was requested:

YES  REOPT(VARS) was specified to reoptimize the access path of the SQL statement at run time.

NO  NOREOPT(VARS) was specified to optimize the access path of the SQL statement only at bind time.

Field Name: QW0112RO

KEEPDYNAMIC  Indicates whether DB2 keeps (KEEPDYNAMIC(YES)) or discards (KEEPDYNAMIC(NO)) prepared SQL statements at commit points.

Field Name: QW0112KD

DBPROTOCOL  Database protocol option. Possible values are:

DRDA

PRIVATE

Field Name: QW0112PR

PREPARE  Indicates whether preparation of dynamic SQL statements was deferred. Possible values are:

YES  Dynamic SQL statement preparation was deferred.

NO  Dynamic SQL statements were prepared immediately.

Field Name: QW0112DP

HINTID  Query optimization hint identifier, the default is blanks.

Field Name: QW0112OH

IMMEDWRITE  Indicates how DB2 updates group buffer pool dependent pages. This is only valid in a data-sharing environment.

Group buffer pool dependent pages can be written out to DASD or SYSTEM pagesets. Values shown are:

NO  DB2 uses normal write activity for updates, this is the default. Pages are written out at, or before phase 2 commit, or at the end of an abort for transactions that have rolled back.

PH1  Pages are written out at, or before phase 1 commit.

If a transaction subsequently rolls back, the pages are updated in the group buffer pool at the end of the rollback, and are written out at the end of the abort.
**IFCID 112 - Thread Allocate**

**YES**  Pages are written out to the coupling facility as soon as the buffer update commits. Pages are written out regardless of whether the update occurs during forward progress or rollback of the transaction.

This option can affect DB2 performance due to coupling facility overhead.

**N/P**  The DB2 subsystem is not part of a data sharing group.

**Field Name:** QW0112WI
Record trace - IFCID 113 - Agent Allocate

The field labels shown in the following sample layout of “Record Trace - IFCID 113 - Agent Allocate” are described in the following section.

<table>
<thead>
<tr>
<th>ALLOCATE</th>
<th>PLAN NAME : BCT</th>
<th>ACQUIRE: ALLOCATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISOLATION  : CS</td>
<td>RELEASE: DEALLOCATION</td>
<td></td>
</tr>
<tr>
<td>DYNAMICRULES: RUN</td>
<td>REOPT : NO</td>
<td></td>
</tr>
<tr>
<td>KEEP_DYNAMIC : NO</td>
<td>PREPARE: NO</td>
<td></td>
</tr>
<tr>
<td>DB_PROTOCOL : PRIVATE</td>
<td>IMMEDWRITNO :</td>
<td></td>
</tr>
<tr>
<td>HINTID : 'BLANK'</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**PLAN NAME**

The plan name for the thread.

Field Name: QW0113PN

**ACQUIRE**

Indicates when to acquire locks:

- **ALLOCATION**
  Acquire the locks when the plan is allocated.
- **USE**
  Acquire the locks when the application first accesses them.

Field Name: QW0113Q

**ISOLATION**

The isolation level:

- **RR**
  Repeatable read
- **CS**
  Cursor stability
- **RS**
  Read stability
- **UR**
  Uncommitted read

Field Name: QW0113I

**RELEASE**

Indicates when to release locks:

- **COMMIT**
  Release locks at commit time.
- **DEALLOCATION**
  Release locks at deallocation time.

Field Name: QW0113R

**DYNAMICRULES**

The value of the DYNAMICRULES option on the BIND/REBIND command:

- **RUN**
  Run-time rules apply to a dynamic SQL statement for authorization checking and object qualification at run time.
- **BIND**
  Bind-time rules apply to a dynamic SQL statement for authorization checking and object qualification at run time.
N/P  DYNAMICRULES was not specified.

Field Name: QW0113DY

REOPT
Indicates whether reoptimization was requested:

YES  REOPT(VARS) was specified to reoptimize the access path of the SQL statement at run time.

NO   NOREOPT(VARS) was specified to optimize the access path of the SQL statement only at bind time.

Field Name: QW0113RO

KEEPDYNAMIC
Indicates whether DB2 keeps (KEEPDYNAMIC(YES)) or discards (KEEPDYNAMIC(NO)) prepared SQL statements at commit points.

Field Name: QW0113KD

DBPROTOCOL
Database protocol option. Possible values are:

DRDA
PRIVATE

Field Name: QW0113PR

PREPARE
Indicates whether preparation of dynamic SQL statements was deferred. Possible values are:

YES   Dynamic SQL statement preparation was deferred.

NO    Dynamic SQL statements were prepared immediately.

Field Name: QW0113DP

HINTID
Query optimization hint identifier, the default is blanks.

Field Name: QW0113OH

IMMEDWRITE
Indicates how DB2 updates group buffer pool dependent pages. This is only valid in a data-sharing environment.

Group buffer pool dependent pages can be written out to DASD or SYSTEM pagesets. Values shown are:

NO  DB2 uses normal write activity for updates, this is the default. Pages are written out at, or before phase 2 commit, or at the end of an abort for transactions that have rolled back.

PH1  Pages are written out at, or before phase 1 commit. If a transaction subsequently rolls back, the pages are updated in the group buffer pool at the end of the rollback, and are written out at the end of the abort.

YES  Pages are written out to the coupling facility as soon as the buffer
IFCID 113 - Agent Allocate

update commits. Pages are written out regardless of whether the update occurs during forward progress or rollback of the transaction.

This option can affect DB2 performance due to coupling facility overhead.

N/P

The DB2 subsystem is not part of a data sharing group.

Field Name: QW0113WI
IFCID 114 - Archive Wait Start

This topic shows detailed information about “Record Trace - IFCID 114 - Archive Wait Start”.

Record trace - IFCID 114 - Archive Wait Start

The field labels shown in the following sample layout of “Record Trace - IFCID 114 - Archive Wait Start” are described in the following section.

DSID 00000002 ACE 2
QW0114HR X'0000' QW0114LR X'00000000'

DSID

The data set identifier of the log manager.

Field Name: QW0114DI

ACE

The relative number of the agent control element address in the ACE cross-reference table. This table is printed at the end of each location for every trace specified.

Field Name: QW0114AC
IFCID 115 - Archive Wait End DASD

IFCID 115 - Archive Wait End DASD

This topic shows detailed information about “Record Trace - IFCID 115 - Archive
Wait End DASD”.

Record trace - IFCID 115 - Archive Wait End DASD

The field labels shown in the following sample layout of “Record Trace - IFCID 115
- Archive Wait End DASD” are described in the following section.

```
RET 0 ACE 1
QW0115BR 0 QW0115BS 0
QW0115FR 0

RET
  The return code.
  Field Name: QW0115RT

ACE
  The relative number of the agent control element address in the ACE
cross-reference table. This table is printed at the end of each location for
every trace specified.
  Field Name: QW0115AC
```
IFCID 116 - Archive Wait End Tape

This topic shows detailed information about “Record Trace - IFCID 116 - Archive Wait End Tape”.

Record trace - IFCID 116 - Archive Wait End Tape

The field labels shown in the following sample layout of “Record Trace - IFCID 116 - Archive Wait End Tape” are described in the following section.

RET 4 ACE 2
QW0116FR 1 QW0116LR 2
QW0116BU X'00000003'

RET

The return code.

Field Name: QW0116RT

ACE

The relative number of the agent control element address in the ACE cross-reference table. This table is printed at the end of each location for every trace specified.

Field Name: QW0116AC
IFCID 117 - Archive Read Start

This topic shows detailed information about “Record Trace - IFCID 117 - Archive Read Start”.

Record trace - IFCID 117 - Archive Read Start

The field labels shown in the following sample layout of “Record Trace - IFCID 117 - Archive Read Start” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>QW0117BR</td>
<td>1</td>
</tr>
<tr>
<td>QW0117RR</td>
<td>3</td>
</tr>
<tr>
<td>QW0117SH</td>
<td>5</td>
</tr>
<tr>
<td>QW0117ER</td>
<td>2</td>
</tr>
<tr>
<td>QW0117ST</td>
<td>4</td>
</tr>
</tbody>
</table>

Field Name: QW0117RT

The request type:

- **RARC**: Read archive request
- **SARC**: Schedule archive read
IFCID 118 - Archive Read End

This topic shows detailed information about “Record Trace - IFCID 118 - Archive Read End”.

Record trace - IFCID 118 - Archive Read End

The field labels shown in the following sample layout of “Record Trace - IFCID 118 - Archive Read End” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>RETURN</td>
<td>1</td>
</tr>
<tr>
<td>QW0118RC</td>
<td>2</td>
</tr>
<tr>
<td>QW0118ST</td>
<td>3</td>
</tr>
<tr>
<td>QW0118SH</td>
<td>4</td>
</tr>
</tbody>
</table>

RETURN

The return code.

Field Name: QW0118RT
This topic shows detailed information about “Record Trace - IFCID 119 - BSDS Write Start”.

Record trace - IFCID 119 - BSDS Write Start

The field labels shown in the following sample layout of “Record Trace - IFCID 119 - BSDS Write Start” are described in the following section.

```
DSID   BSDS0001  ACE   1
QW0119HR  X'0000'  QW0119LR  X'0A000001'
```

**DSID**

The data set identifier.

**Field Name:** QW0119DI

**ACE**

The relative number of the agent control element address in the ACE cross-reference table. This table is printed at the end of each location for every trace specified.

**Field Name:** QW0119AC
IFCID 120 - BSDS Write End

This topic shows detailed information about “Record Trace - IFCID 120 - BSDS Write End”.

**Record trace - IFCID 120 - BSDS Write End**

The field labels shown in the following sample layout of “Record Trace - IFCID 120 - BSDS Write End” are described in the following section.

**RETURN**

The return code.

**Field Name:** QW0120RT

**ACE**

The relative number of the agent control element address in the ACE cross-reference table. This table is printed at the end of each location for every trace specified.

**Field Name:** QW0120AC
IFCID 121 - IBM Service Record

This topic shows detailed information about “Record Trace - IFCID 121 - IBM Service Record”.

This record is for IBM service use.
This topic shows detailed information about “Record Trace - IFCID 122 - IBM Service Record”.

This record is for IBM service use.
IFCID 123 - SRV Record

IFCID 123 - SRV Record

This topic shows detailed information about “Record Trace - IFCID 123 - SRV Record”.

Record trace - IFCID 123 - SRV Record

The field labels shown in the following sample layout of “Record Trace - IFCID 123 - SRV Record” are described in the following section.

LENGTH

The length of the area.

Field Name: QW0123LN

IFCID

The IFCID of the original entry.

Field Name: QW0123ID
Record trace - IFCID 124 - SQL Statement Record

The field labels shown in the following sample layout of “Record Trace - IFCID 124 - SQL Statement Record” are described in the following section.

**CONNECTION TYPE**

The connection type. 'BLANK' indicates that there is no connection type.

Field Name: QW01246Y

**ACE TOKEN**

The ACE token in hexadecimal.

Field Name: QW01246A

**THREAD ASID**

The ASID of the thread in hexadecimal.

Field Name: QW01246S

**ASCB TOKEN**

The ASCB token in hexadecimal.

Field Name: QW01246C

**APPL REQ COUNT**

The number of calls to DB2 in decimal.

Field Name: QW01246Q

**AGENT NAME**

When an agent is running a stored procedure, trigger, or user-defined function, this is the unqualified name of the routine. Otherwise, this field is blank.

Field Name: QW01246P

**THREAD STATUS**

The status of the thread.

Field Name: QW01246I

**TCB TOKEN**

The TCB token in hexadecimal.

Field Name: QW01246T

**THREAD TYPE**

The type of thread.
IFCID 124 - SQL Statement Record

Field Name: QW0124D

AGENT TYPE
The type of the agent.
Field Name: QW0124AF

STATEMENT TYPE
The statement type.
Field Name: QW0124TK

STATEMENT IDENTIFIER
The statement ID (former Cached Dynamic SQL Identifier).
Field Name: QW0124ST

LOCATION
The name of the location where the thread executes the package.
Field Name: QW0124LN

COLLECTION
The collection name.
Field Name: QW0124CI

PACKAGE
The package identifier.
Field Name: QW0124PN

TOKEN
The consistency token.
Field Name: QW0124CN

NETWORKID
The network identifier.
Field Name: QW0124NI

LUNAME
The logical unit name.
Field Name: QW0124LM

UNIQUENESS VALUE
The instance number.
Field Name: QW0124UV

LUWSEQ
The LUW sequence number.
Field Name: QW0124CC

ENDUSER
The user ID at the end user's workstation.
Field Name: QW0124EI
WSNAME

The workstation name of the end user.

Field Name: QW0124EW

TRANSACT

The transaction name of the end user.

Field Name: QW0124ET
IFCID 125 - RID Pool Processing

IFCID 125 - RID Pool Processing

This topic shows detailed information about “Record Trace - IFCID 125 - RID Pool Processing”.

This record is written when performance class 8 is ON. Monitor privilege is required for reading via IFI. The record contains standard information and one section for each index used to obtain candidate record identifiers (RIDs).

Record trace - IFCID 125 - RID Pool Processing

The field labels shown in the following sample layout of “Record Trace - IFCID 125 - RID Pool Processing” are described in the following section.

RID POOL NETWORKID: DEIBMIPS LUNAME: IPSATDA2 LUWSEQ: 1
PROCESSING COLLECTION : COLLECTIONID000001
PLAN NAME : PLANID001
PROGRAM NAME: PROGRAMNAME0000001
CONSISTENCY TOKEN: X'D4C9C4D5C9C7C8E3'
USED: YES NOT USED: N/A
RIDS IN FINAL LIST: 21

COLLECTION
Package collection identifier for this query.
Field Name: QW0125PC

PLAN NAME
Plan name for this query.
Field Name: QW0125PL

PROG NAME
Program name for this query.
Field Name: QW0125PN

STATEMENT NUMB
Statement number for this query.
Field Name: QW0125SN

CONSISTENCY TOKEN
Consistency token for this query.
Field Name: QW0125TS

USED
Indicates whether multiple index access paths are used, or whether RID pool processing is invoked.
Field Name: QW0125AT

NOT USED
Indicates why multiple index access paths are not used, or whether RID pool processing is not invoked.
Field Name: QW0125MR

RIDS IN FINAL LIST

40-470  OMEGAMON XE for DB2 PE & PM: Report Reference
**IFCID 125 - RID Pool Processing**

The number of record identifiers in the final index list. It indicates how many RID sections are printed. Each RID section contains one set of DBID, INDEX RID, OBID, and THRESHOLD data.

This field can also contain NO STORAGE, or MAX EXCEEDED.

**Field Name:** QW0125NR

**DBID**

The database ID. Deduced from the DB2 fields QW0125DB, QW0105DN or QW0107DN.

When present, the database name is shown, otherwise the decimal identifier from QW0125DB is shown, or N/A when this value is 0.

**Field Name:** RT0125DB

**INDEX RIDS**

The number of record identifiers in the index. This field can also contain one of the following values:

- NO RETRIEVAL
- NO STORAGE
- LIMIT EXCEEDED
- N/P

**Field Name:** QW0125RI

**OBID**

The object ID. Deduced from the DB2 fields QW0125OB, QW0105TN or QW0107TN.

When present, the name of the object is shown, otherwise the decimal identifier from QW0125OB is shown, or N/A when this value is 0.

**Field Name:** RT0125OB

**THRESHOLD**

The highest value of RIDs allowed for this index.

**Field Name:** QW0125TH
IFCID 126 - Log Buffer Write

This topic shows detailed information about “Record Trace - IFCID 126 - Log Buffer Write”.

Record trace - IFCID 126 - Log Buffer Write

The field labels shown in the following sample layout of “Record Trace - IFCID 126 - Log Buffer Write” are described in the following section.
**IFCID 127 - Page Wait I/O In Prog (Start)**

This topic shows detailed information about “Record Trace - IFCID 127 - Page Wait I/O In Prog (Start)”.

**Record trace - IFCID 127 - Page Wait I/O In Prog (Start)**

The field labels shown in the following sample layout of “Record Trace - IFCID 127 - Page Wait I/O In Prog (Start)” are described in the following section.

```
DBID: DSNDB06  OBID: SYSDBASE
PAGE NUMBER: X'001ED0'  TYPE OF I/O: READ
POOL ID: 0  ACE: 1
TABLE SPACE TYPE: L
```

**DBID**

The database ID. Deduced from the DB2 fields QW0127DB, QW0105DN or QW0107DN.

When present, the database name is shown, otherwise the decimal identifier from QW0127DB is shown, or N/A when this value is 0.

Field Name: RT0127DB

**OBID**

The object ID. Deduced from the DB2 fields QW0127OB, QW0105TN or QW0107TN.

When present, the name of the object is shown, otherwise the decimal identifier from QW0127OB is shown, or N/A when this value is 0.

Field Name: RT0127OB

**PAGE NUMBER**

The number of the page being read or written.

Field Name: QW0127PN

**TYPE OF I/O**

The type of I/O process.

Field Name: QW0127F

**POOL ID**

The buffer pool internal identifier. The values 0 through 49 are the identifiers for BP0 through BP49. The values 80 through 89 are the identifiers for BP32K through BP32K9.

Field Name: QW0127BP

**ACE**

The relative number of the agent control element address in the ACE cross-reference table. This table is printed at the end of each location for every trace specified.

Field Name: QW0127AC

**TABLE SPACE TYPE**

The type of the table space:

L Non-EA large table
IFCID 127 - Page Wait I/O In Prog (Start)

N  Non-large table
V  EA-enabled large table

Field Name: QW0127FG
This topic shows detailed information about “Record Trace - IFCID 128 - Page Wait I/O In Prog (End)".

Record trace - IFCID 128 - Page Wait I/O In Prog (End)

The field labels shown in the following sample layout of “Record Trace - IFCID 128 - Page Wait I/O In Prog (End)” are described in the following section.

DBID: DSND806
OBID: SYSDBASE
PAGE NUMBER: X'001ED0'
TYPE OF I/O: READ
POOL ID: 0
ACE: 1
TABLE SPACE TYPE: L

**DBID**

The database ID. Deduced from the DB2 fields QW0128DB, QW0105DN or QW0107DN.

When present, the database name is shown, otherwise the decimal identifier from QW0128DB is shown, or N/A when this value is 0.

**Field Name:** RT0128DB

**OBID**

The object ID. Deduced from the DB2 fields QW0128OB, QW0105TN or QW0107TN.

When present, the name of the object is shown, otherwise the decimal identifier from QW0128OB is shown, or N/A when this value is 0.

**Field Name:** RT0128OB

**PAGE NUMBER**

The number of the page being read or written.

**Field Name:** QW0128PN

**TYPE OF I/O**

The type of I/O process.

**Field Name:** QW0128F

**ACE**

The relative number of the agent control element address in the ACE cross-reference table. This table is printed at the end of each location for every trace specified.

**Field Name:** QW0128AC

**STATUS FL**

The status flag indicating whether the I/O process was canceled.

**Field Name:** QW0128S

**TABLE SPACE TYPE**

The type of the table space:

- **L** Non-EA large table
- **N** Non-large table
- **V** EA-enabled large table
IFCID 128 - Page Wait I/O In Prog (End)

Field Name: QW0128FG
Record trace - IFCID 129 - CI-S Obtained via IFI Reads

The field labels shown in the following sample layout of “Record Trace - IFCID 129 - CI-S Obtained via IFI Reads” are described in the following section.
IFCID 140 - Audit Auth Failures

This topic shows detailed information about “Record Trace - IFCID 140 - Audit Auth Failures”.

Record trace - IFCID 140 - Audit Auth Failures

The field labels shown in the following sample layout of “Record Trace - IFCID 140 - Audit Auth Failures” are described in the following section.

AUTH CHECKED : DB2PM
REASON : 0 STATMNT LENGTH: 65532
PRIV CHECKED : CREATE TABLE RETCOD: -1
OBJECT: DATABASE OPTIONS: X'0400000000000000'
SOURCE OBJECT: DB2PM
SOURCE OWNER : DB2PM
TARGET OBJECT: GRANT
TARGET OWNER : DB2PM
SQL STMT:
ACEE UTOKEN : 'BLANK'
RID OF ROW : 'BLANK' SECLABEL OF ROW: 'BLANK'
AUTH ID TYPE : PRIMARY OR SECONDARY AUTH ID

AUTH CHECKED

The authorization ID being checked.

Field Name: QW0140UR

REASON

The user-defined reason code from the access control authorization exit routine.

Field Name: QW0140RS

STATMNT LENGTH

Is the length of the failing SQL statement plus 4. It is zero (0) if no SQL statement exists.

Field Name: QW0140LL

PRIV CHECKED

The privilege that was checked. Possible values are provided in the DB2 macro DSNDQW02.

Field Name: QW0140PR

RETCOD

The return code from the access control authorization exit routine.

Field Name: QW0140RC

OBJECT

The object type. N/P is printed if there is no object type. Possible values are:

- ACEE
- APPLICATION PLAN
- BUFFERPOOL
- COLLECTION
- DATABASE
IFCID 140 - Audit Auth Failures

- DISTINCT TYPE
- FUNCTION
- JAR
- PACKAGE
- PROCEDURE
- ROW
- SCHEMA
- SEQUENCE
- STORAGE GROUP
- TABLE OR VIEW
- TABLESPACE
- USER AUTH
  System privileges, such as SYSADM or SYSOPR

Field Name: QW0140OB

OPTIONS

The options used in the host to check the SQL statement. The bits of this field are used as indicators. If all bits are 0, the statement is not an SQL statement. The values are:

Bit 1  Host language character string delimiter
       0  Apostrophe
       1  Quote

Bit 2  Decimal point symbol
       0  Period
       1  Comma

Bit 3  SQL character string delimiter
       0  Apostrophe
       1  Quote

Bit 4  Mixed character string indicator
       0  No
       1  Yes

Bit 5  Host language options indicator
       0  Do not use host language options
       1  Use host language options

Bits 6 to 8  Host language indicator
       001  Assembler
       010  Cobol
       011  PL/I
       100  None - Dynamic SQL
       101  Fortran
### IFCID 140 - Audit Auth Failures

110  Cobol2  
111  Null - See bits 17 to 24 for the language  

#### Bits 9 to 16  
Character set being used  
- 00000000  Alphanumeric  
- 00000001  Katakana  

#### Bits 17 to 24  
Alternate host language field  
- B  Assembler  
- C  Cobol  
- P  PL/I  
- F  Fortran  
- 2  Cobol2  
- D  C  

#### Bits 25 to 28  
Time option  
- 0000  None  
- 1000  Local  
- 0100  JIS  
- 0010  ISO/EUR  
- 0001  USA  

#### Bits 29 to 32  
Date option  
- 0000  None  
- 1000  Local  
- 0100  EUR  
- 0010  ISO/JIS  
- 0001  USA  

#### Bit 33  
Decimal  
- 0  No  
- 1  Yes  

#### Bits 34 to 40  
Unused  

#### Bits 41 to 48  
Remote option  
- 00000001  SQL(ALL)  
- 00000010  SQL(DB2)
IFCID 140 - Audit Auth Failures

Bits 49 to 56
   SQL flag option
      00000000   No SQLFLAG option
      00000001   SQLFLAG(SAA)

Field Name: QW0140HO
SOURCE OBJECT
   The source object name.
   Field Name: QW0140SN

SOURCE OWNER
   The source object owner.
   Field Name: QW0140SC

TARGET OBJECT
   The target object name.
   Field Name: QW0140TN

TARGET OWNER
   The target object owner.
   Field Name: QW0140TC

SQL STMT
   The SQL statement text. Long SQL text can be truncated.
   Field Name: QW0140SQ

ACEE UTOKEN
   Shows the ACEE UTOKEN, if it is available. If it is not available, the first
   word of this field contains one of the following values:
      0      The UTOKEN cannot be accessed
      -1     An abend occurred during the attempt to access the ACEE.

Field Name: QW0140UT

RID OF ROW
   Shows the RID of the row that is updated or deleted if the table has
   multilevel security.
   Field Name: QW0140ID

SECLABEL OF ROW
   Shows the security label of the row, for a table with multilevel security.
   Field Name: QW0140RL

AUTH ID TYPE
   The authorization ID type. Possible values are:
      L      A ROLE is used.
IFCID 140 - Audit Auth Failures

*blank*  The user ID of the primary or the secondary authorization ID is used.

N/P  A blank is shown in the performance database.

N/A  A blank is shown in the performance database.

**Field Name:** QW0140AT
IFCID 141 - Audit DDL Grant/Revoke

This topic shows detailed information about “Record Trace - IFCID 141 - Audit DDL Grant/Revoke”.

Record trace - IFCID 141 - Audit DDL Grant/Revoke

The field labels shown in the following sample layout of “Record Trace - IFCID 141 - Audit DDL Grant/Revoke” are described in the following section.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GRANTOR/REVOKER</td>
<td>The authorization ID of the user (grantor/revoker) who received access.</td>
</tr>
<tr>
<td></td>
<td>Possible values:</td>
</tr>
<tr>
<td>L</td>
<td>A ROLE is used.</td>
</tr>
<tr>
<td>blank</td>
<td>The user ID of the primary or the secondary authorization ID is used.</td>
</tr>
<tr>
<td>N/P</td>
<td>A blank is shown in the performance database.</td>
</tr>
<tr>
<td>N/A</td>
<td>A blank is shown in the performance database.</td>
</tr>
<tr>
<td>Field Name: QW0141OR</td>
<td></td>
</tr>
</tbody>
</table>

REASON

The reason why access was granted.

In the Audit report set this field is only valid for GRANTs. It indicates the authorization level of the grantor. For REVOKEs and unsuccessful GRANTs, N/A is printed.

Possible values are:

- PACKADMA (abbreviation for PACKADM ON ALL COLLECTIONS)
- DBCTRL
- DBADM
- SECADM
- ACCCTRL (abbreviation for ACCESSCTRL)
- SYSCTRL
- DBMAINT
- SYSOPR
- PACKADMS (abbreviation for PACKADM ON A SPECIFIC COLLECTION-ID)
- SYSADM

Field Name: QW0141RE

RETURN

The SQL return code.

Chapter 40. IFCID Record Blocks 40-483
IFCID 141 - Audit DDL Grant/Revoke

Field Name: QW0141CO

OBJECT
The object type. Possible values are:

BUFFERPOOL
Buffer Pool

COLLECTION
Collection

DATABASE
Database

DISTINCT TYPE
Distinct Type

FUNCTION
Function

PACKAGE
Package

SCHEMA
Schema

PROCEDURE
Procedure

APPLICATION PLAN
Application Plan

TABLESPACE
Table Space

STORAGE GROUP
Storage Group

TABLE OR VIEW
Table or View

USER AUTH
System privileges, such as SYSADM or SYSOPR

Field Name: QW0141OB

OPTIONS
The options used in the host to check the SQL statement. The bits of this field are used as indicators. If all bits are 0, the statement is not an SQL statement. The values are:

Bit 1  Host language character string delimiter
       0  Apostrophe
       1  Quote

Bit 2  Decimal point symbol
       0  Period
       1  Comma

Bit 3  SQL character string delimiter
       0  Apostrophe
IFCID 141 - Audit DDL Grant/Revoke

1 Quote

**Bit 4** Mixed character string indicator
0 No
1 Yes

**Bit 5** Host language options indicator
0 Do not use host language options
1 Use host language options

**Bits 6 to 8** Host language indicator
001 Assembler
010 Cobol
011 PL/I
100 None - Dynamic SQL
101 Fortran
110 Cobol2
111 Null - See bits 17 to 24 for the language

**Bits 9 to 16** Character set being used
00000000 Alphanumeric
00000001 Katakana

**Bits 17 to 24** Alternate host language field
B Assembler
C Cobol
P PL/I
F Fortran
2 Cobol2
D C

**Bits 25 to 28** Time Option
0000 None
1000 Local
0100 JIS
0010 ISO/EUR
0001 USA

**Bits 29 to 32** Date Option
0000 None
IFCID 141 - Audit DDL Grant/Revoke

1000  Local
0100  EUR
0010  ISO/JIS
0001  USA

Bit 33  Decimal
0  No
1  Yes

Bits 34 to 40  Unused

Bits 41 to 48  Remote option
00000001  SQL(ALL)
00000010  SQL(DB2)

Bits 49 to 56  SQL flag option
00000000  No SQLFLAG option
00000001  SQLFLAG(SAA)

Field Name: QW0141HO

GRANTOR TYPE/REVOKER TYPE

The authorization ID of the owner. Possible values are:

A ROLE or ROLE
A role is used.

PRIMARY OR SECONDARY AUTH ID or PRIM/SECOND AUTHID
The user ID of the primary or the secondary authorization ID is used.

N/P or N/A
The field is not present or not applicable. String NONE is shown in the performance database.

N/A A blank is shown in the performance database.

Field Name: QW0141OT

SQL STMT

The SQL statement text. Long SQL text can be truncated.

Field Name: QW0141TX
This topic shows detailed information about “Record Trace - IFCID 142 - Audit DDL Create/Alter/Drop”.

Audit DDL reports on SQL CREATE, ALTER, and DROP statements executed against an auditable object.

The SQL statement types are AUDIT DDL CREATE, AUDIT DDL ALTER, or AUDIT DDL DROP. These statements are all reported in the same format.

**Record trace - IFCID 142 - Audit DDL Create/Alter/Drop**

The field labels shown in the following sample layout of “Record Trace - IFCID 142 - Audit DDL Create/Alter/Drop” are described in the following section.

```
AUDIT DDL  NETWORKID: DEIBMIPS LUNAME: IPSAU851 LUWSEQ: 1
UNKNOWN   TABLE NAME : AUDTB1
TABLE OWNER : PRL
TABLE CREATOR: PRL
TABLE OWNER TYPE:N/A
OPTIONS : X'0400000000000000'
DATABASE : 264
TABLE OBID : 3
SECLABEL OF MLS TABLE: N/P
MULTILEVEL SECURITY : N/P
ROW/CLMN ACCESS CTRL : B
SQL STMT:
CREATE TABLE PRL.AUDTB1 (IDCOLUMN INTEGER
GENERATED ALWAYS AS IDENTITY, NNAME VARCHAR
(50) NOT NULL, VNAME CHAR(10) NOT NULL,
ANZAHL INTEGER NOT NULL) AUDIT ALL IN
```

**TABLE NAME**

The table name being created, altered, or dropped.

*Field Name:* QW0142TN

**TABLE OWNER**

The table owner (same as table qualifier).

*Field Name:* QW0142OW

**TABLE CREATOR**

The table creator.

*Field Name:* QW0142CR

**TABLE OWNER TYPE**

The type of the table owner (grantor or revoker). Possible values are:

- **L** A ROLE is used.
- **blank** The user ID of the primary or the secondary authorization ID is used.
- **N/P** A blank is shown in the performance database.
- **N/A** A blank is shown in the performance database.

*Field Name:* QW0142OR

**OPTIONS**
IFCID 142 - Audit DDL Create/Alter/Drop

The options used in the host to check the SQL statement. The bits of this field are used as indicators. If all bits are 0, the statement is not an SQL statement. The values are:

**Bit 1**  Host language character string delimiter
   0     Apostrophe
   1     Quote

**Bit 2**  Decimal point symbol
   0     Period
   1     Comma

**Bit 3**  SQL character string delimiter
   0     Apostrophe
   1     Quote

**Bit 4**  Mixed character string indicator
   0     No
   1     Yes

**Bit 5**  Host language options indicator
   0     Do not use host language options
   1     Use host language options

**Bits 6 to 8**  Host language indicator
   001     Assembler
   010     Cobol
   011     PL/I
   100     None - Dynamic SQL
   101     Fortran
   110     Cobol2
   111     Null - See bits 17 to 24 for the language

**Bits 9 to 16**  Character set being used
   00000000     Alphanumeric
   00000001     Katakana

**Bits 17 to 24**  Alternate host language field
   B     Assembler
   C     Cobol
   P     PL/I
   F     Fortran
   2     Cobol2
IFCID 142 - Audit DDL Create/Alter/Drop

Bits 25 to 28
Time Option
0000  None
1000  Local
0100  JIS
0010  ISO/EUR
0001  USA

Bits 29 to 32
Date Option
0000  None
1000  Local
0100  EUR
0010  ISO/JIS
0001  USA

Bit 33  Decimal
0   No
1   Yes

Bits 34 to 40
Unused

Bits 41 to 48
Remote option
00000001  SQL(ALL)
00000010  SQL(DB2)

Bits 49 to 56
SQL flag option
00000000  No SQLFLAG option
00000001  SQLFLAG(SAA)

Field Name: QW0142HO

DATABASE
The database ID. Deduced from the DB2 fields QW0142DB, QW0105DN or QW0107DN.
When present, the database name is shown, otherwise the decimal identifier from QW0142DB is shown, or N/A when this value is 0.

Field Name: RT0142DB

TABLE OBID
ICCID 142 - Audit DDL Create/Alter/Drop

The object ID. Deduced from the DB2 fields QW0142OB, QW0105TN or QW0107TN.

When present, the name of the object is shown, otherwise the decimal identifier from QW0142OB is shown, or N/A when this value is 0.

Field Name: RT0142OB

SECLABEL OF MLS TABLE

The security label that is used when the table is defined.

Field Name: QW0142SL

MULTILEVEL SECURITY

The Multilevel Security (MLS) table can contain the following values:

YES For a Create or Drop operation of a table that has multilevel security, or for an Alter operation of a table to add a security label column.

NO For an Alter operation of a table that has multilevel security.

NON MLS TABLE

The table does not have multilevel security.

N/P Not present. A blank is shown in the performance database.

N/A A blank is shown in the performance database.

Field Name: QW0142ML

ROW/CLMN ACCESS CTRL

The access control field contains data about ROW-LEVEL and COLUMN-LEVEL (R/C) ACCESS CONTROL in DDL. It can have the following values:

'R' (ROW)
Activates row-level access control.

'C' (COLUMN)
Activates column-level access control.

'B' (BOTH)
Activates row-level and column-level access control.

' ' (NONE)
Activates no access control.

Field Name: QW0142RC

SQL STMT

The SQL statement text. Long SQL text can be truncated.

Field Name: QW0142TX
IFCID 143 - Audit First Write

This topic shows detailed information about “Record Trace - IFCID 143 - Audit First Write”.

Record trace - IFCID 143 - Audit First Write

The field labels shown in the following sample layout of “Record Trace - IFCID 143 - Audit First Write” are described in the following section.

DATABASE: DSN8D23A    LOGRBA: X'000000000000'
PAGE SET: 4    TABLE OBID: 14

DATABASE

The database ID. Deduced from the DB2 fields QW0143DB, QW0105DN or QW0107DN.

When present, the database name is shown, otherwise the decimal identifier from QW0143DB is shown, or N/A when this value is 0.

Field Name: RT0143DB

LOGRBA

The identifier of the unit of recovery.

Field Name: QW0143UR

PAGESET

The page set name or decimal identifier.

Field Name: RT0143PS

TABLE OBID

The object ID. Deduced from the DB2 fields QW0143OB, QW0105TN or QW0107TN.

When present, the name of the object is shown, otherwise the decimal identifier from QW0143OB is shown or N/A when this value is 0.

Field Name: RT0143OB
IFCID 144 - Audit First Read

IFCID 144 - Audit First Read

This topic shows detailed information about “Record Trace - IFCID 144 - Audit First Read”.

Record trace - IFCID 144 - Audit First Read

The field labels shown in the following sample layout of “Record Trace - IFCID 144 - Audit First Read” are described in the following section.

DATABASE: DSNDB06 LOGRBA: X'000000000000'
PAGE SET: DSNDSX01 TABLE OBID: 5

DATABASE

The database ID. Deduced from the DB2 fields QW0144DB, QW0105DN or QW0107DN.

When present, the database name is shown, otherwise the decimal identifier from QW0144DB is shown, or N/A when this value is 0.

Field Name: RT0144DB

LOGRBA

The identifier of the unit of recovery.

Field Name: QW0144UR

PAGESET

The page set name or decimal identifier.

Field Name: QW0144PS
IFCID 145 - Audit DML Statement

This topic shows detailed information about “Record Trace - IFCID 145 - Audit DML Statement”.

Record trace - IFCID 145 - Audit DML Statement

The field labels shown in the following sample layout of “Record Trace - IFCID 145 - Audit DML Statement” are described in the following section.

--- Audit Log Record ---

SECTION 1 - Audit Information:

LOCATION NAME: DB0B
PKG COLLCT ID: DSNTEP2
PROGRAM NAME: DSNTEP2
HOST OPTIONS: X'0400000000000000'
SQL CODE: 0
STMT # : 1829
LOCATION NAME (LONG): DB0B
PKG COLLCT ID (LONG): DSNTEP2
PROGRAM NAME (LONG): DSNTEP2

SECTION 2 - Audit Objects:

DATABASE: 353
TABLE OBID: 3

SECTION 3 - Row/Column Access Ctrl Objects:

SCHEMA NAME: DB0BSECA
OBJECT NAME: INCOME_BRANCH
SCHEMA NAME: DB0BSECA
OBJECT NAME: RA01_CUSTOMERS

SECTION 4 - SQL Statement Text:

SELECT * FROM DB2R5.CUSTOMER

LOCATION NAME
The location name.
Field Name: QW0145LN

PKG COLLCT ID
The package collection identifier.
Field Name: QW0145PC

PROGRAM NAME
The program name.
Field Name: QW0145PN

STMT TIME
The hexadecimal value of the precompiler timestamp.
Field Name: QW0145TS

TYPE
The SQL statement type.
Field Name: QW0145ST

ISOLATION
The isolation level of the DML statement:
RR Repeateable read
IFCID 145 - Audit DML Statement

Field Name: QW0145IS

HOST OPTIONS
The options used in the host to check the SQL statement. The bits of this field are used as indicators. If all bits are 0, the statement is not an SQL statement. The values are:

**Bit 1** Host language character string delimiter
- 0  Apostrophe
- 1  Quote

**Bit 2** Decimal point symbol
- 0  Period
- 1  Comma

**Bit 3** SQL character string delimiter
- 0  Apostrophe
- 1  Quote

**Bit 4** Mixed character string indicator
- 0  No
- 1  Yes

**Bit 5** Host language options indicator
- 0  Do not use host language options
- 1  Use host language options

**Bits 6 to 8**
Host language indicator
- 001  Assembler
- 010  Cobol
- 011  PL/I
- 100  None - Dynamic SQL
- 101  Fortran
- 110  Cobol2
- 111  Null - See bits 17 to 24 for the language

**Bits 9 to 16**
Character set being used
- 00000000  Alphanumeric
IFCID 145 - Audit DML Statement

Bits 17 to 24
Alternate host language field

B   Assembler
C   Cobol
P   PL/I
F   Fortran
2   Cobol2
D   C

Bits 25 to 28
Time Option

0000  None
1000  Local
0100  JIS
0010  ISO/EUR
0001  USA

Bits 29 to 32
Date Option

0000  None
1000  Local
0100  EUR
0010  ISO/JIS
0001  USA

Bit 33  Decimal
0   No
1   Yes

Bits 34 to 40
Unused

Bits 41 to 48
Remote option

00000001  SQL(ALL)
00000010  SQL(DB2)

Bits 49 to 56
SQL flag option

00000000  No SQLFLAG option
00000001  SQLFLAG(SAA)
IFCID 145 - Audit DML Statement

Field Name: QW0145HO

SQL CODE
The SQLCODE of the SQL statement.
Field Name: QW0145SC

STMT #
The precompiler statement number.
Field Name: QW0145SN

STMT ID
The SQL unique statement ID.
Field Name: QW0145SI

DBID/OBID #
The number of unique non-zero DBID and OBID in the audited statement.
Field Name: QW0145OB_NUM

MASK/PERM #
The number of masks or permissions enforced in the audited statement.
Field Name: QW0145AC_NUM

LOCATION NAME (LONG)
The location name.
Field Name: QW0145LN

PKG COLLECT ID (LONG)
The package collection identifier.
Field Name: QW0145PC

PROGRAM NAME (LONG)
The program name.
Field Name: QW0145PN

DATABASE
The audit log table DBID in hexadecimal.
Field Name: QW0145DB

TABLE OBID
The audit log table OBID in hexadecimal.
Field Name: QW0145OB

SCHEMA NAME
The name of the access control schema.
Field Name: QW0145AS

OBJECT NAME
The name of the access control object.
Field Name: QW0145AO
SECTION 4 - SQL STATEMENT TEXT (NO LABEL)

The SQL statement text associated with the BIND. If SQL text is not present, N/P is printed. Long SQL text can be truncated.

Field Name: QW0145RT
IFCID 146 - User Record

This topic shows detailed information about “Record Trace - IFCID 146 - User Record”.

When present, the IFCID 146 record is printed in the standard hexadecimal dump format. The character format is on the right.
IFCID 147 - Thread Summary

This record only contains data from an Online Monitor trace data set.

"IFCID 147 - Data Sharing Accounting Data" on page 40-500
This topic shows detailed information about “Record Trace - IFCID 147 - Data Sharing Accounting Data”.

"IFCID 147 - Distributed Header Data" on page 40-501
This topic shows detailed information about “Record Trace - IFCID 147 - Distributed Header Data”.

"IFCID 147 - Instrumentation Accounting Data" on page 40-503
This topic shows detailed information about “Record Trace - IFCID 147 - Instrumentation Accounting Data”.

"IFCID 147 - Instrumentation Accounting Data Overflow” on page 40-504
This topic shows detailed information about “Record Trace - IFCID 147 - Instrumentation Accounting Data Overflow”.

"IFCID 147 - Logging” on page 40-507
This topic shows detailed information about “Record Trace - IFCID 147 - Logging”.

"IFCID 147 - Monitor Detail Data” on page 40-508
This topic shows detailed information about “Record Trace - IFCID 147 - Monitor Detail Data”.

"IFCID 147 - Thread Correlation Data” on page 40-514
This topic shows detailed information about “Record Trace - IFCID 147 - Thread Correlation Data”.

Record trace - IFCID 147 - Data Sharing Accounting Data

The field labels shown in the following sample layout of “Record Trace - IFCID 147 - Data Sharing Accounting Data” are described in the following section.

DATA SHARING ACCOUNTING DATA
MEMBER NAMES: N/P

MEMBER NAMES

For an assisting task, the name of the parallelism coordinator. For a coordinating task, the name of each assisting member.

Field Name: QWDAXCQO
IFCID 147 - Distributed Header Data

This topic shows detailed information about “Record Trace - IFCID 147 - Distributed Header Data”.

Place text here

Record trace - IFCID 147 - Distributed Header Data

The field labels shown in the following sample layout of “Record Trace - IFCID 147 - Distributed Header Data” are described in the following section.

DISTRIBUTED HEADER DATA
REQUESTING LOCATION : REQUESTLOCATION1
AR NAME : SRVNAMPARAMETER1
REQUESTING TIMESTAMP : N/P PRDID : PRD VID RD M1

REQUESTING LOCATION

The location name of the requester. If the thread is an allied thread (no distributed requests) or the thread is an allied-distributed thread (this location is the requester), OMEGAMON XE for DB2 PE sets this field equal to the local location. If the thread is a database access thread (this location is a server).

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup. For parallel query rollup records, the value will be derived from the parent record.

Field Name: QWHDRQNM

AR NAME

The application requester name.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup. For parallel query rollup records, the value will be derived from the parent record.

Field Name: QWHDSVNM

REQUESTING TIMESTAMP

The timestamp for database access thread (DBAT) records.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup. For parallel query rollup records, the value will be derived from the parent record.

Field Name: QWHDTSTP

PRDID

Shows the product identifier (ID) of the requester. It can have the following values:

DB2 For DB2 UDB for z/OS

SQL/DS For DB2 UDB for VSE and VM

JDBC DRIVER For Universal JDBC driver

COMMON SERV For DB2 UDB for Linux, UNIX, Windows
IFCID 147 - Distributed Header Data

DB2/400
   For DB2 UDB for iSeries

Otherwise, it shows the first 3 characters of the product ID, or N/P if the record was written at the application requester location.

For DDF/RRSAF rollup records, the field will contain a value derived from the last thread to rollup. For parallel query rollup records, the value will be derived from the parent record.

Field Name: QWHDPRID
IFCID 147 - Instrumentation Accounting Data

This topic shows detailed information about “Record Trace - IFCID 147 - Instrumentation Accounting Data”.

Note: This report has the same layout as IFCID 003, for details refer to “IFCID 003 - Instrumentation Accounting Data” on page 40-175.
IFCID 147 - Instrumentation Accounting Data Overflow

This topic shows detailed information about “Record Trace - IFCID 147 - Instrumentation Accounting Data Overflow”.

Record trace - IFCID 147 - Instrumentation Accounting Data Overflow

The field labels shown in the following sample layout of “Record Trace - IFCID 147 - Instrumentation Accounting Data Overflow” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARCH.LOG(QUIES) SUSP TIME</td>
<td>The accumulated waiting time due to the processing of ARCH LOG MODE(QUIESCE) commands. This time does not represent the time required to perform the entire command. Field Name: QWAXALOG This is an exception field.</td>
</tr>
<tr>
<td>ARCH.LOG READ SUSP TIME</td>
<td>The accumulated waiting time due to the processing of ARCH LOG READ SUSP EVENTS.</td>
</tr>
<tr>
<td>DRAIN LOCK SUSP TIME</td>
<td>The accumulated waiting time for a drain lock. This is the time the requester is suspended while waiting to acquire the drain lock. Field Name: QWAXAWDR This is an exception field.</td>
</tr>
<tr>
<td>CLAIM RELEASE SUSP TIME</td>
<td>The accumulated waiting time for a claim to be released. After the drain lock is acquired, the drainer must wait for claim holders to release the object. Field Name: QWAXAWCL This is an exception field.</td>
</tr>
<tr>
<td>COMMIT PH1 WRITE I/O TIME</td>
<td>The number of wait trace events processed for waits for drain locks. Field Name: QWAXARND</td>
</tr>
<tr>
<td>ASYNCH. IXL REQ. TIME</td>
<td>The number of wait trace events processed for waits for claims to be released.</td>
</tr>
</tbody>
</table>
IFCID 147 - Instrumentation Accounting Data Overflow

Field Name: QWAXARNC

OPEN/CLOSE SUSP TIME

Accumulated waiting time for a synchronous execution unit switch to the DB2 OPEN/CLOSE data set service for the HSM recall service.

This value is an average.

Field Name: QWAXOCSE

OPEN/CLOSE SUSP EVENTS

Number of wait trace events processed of waits for synchronous execution unit switching to the Open/Close service.

Field Name: QWAXOCNS

SYSLGRNG SUSP TIME

Accumulated wait time for a synchronous execution unit switch to the DB2 SYSLGRNG recording service. This service is sometimes used for Level ID checking for downlevel detection.

This value is an average.

Field Name: QWAXSLSE

SYSLGRNG SUSP EVENTS

Number of wait trace events for a synchronous execution unit switch to the DB2 SYSLGRNG recording service.

Field Name: QWAXSLNS

EXC/DEL/DEF SUSP TIME

Accumulated wait time for a synchronous execution unit switch to the DB2 data space manager services. This includes DEFINE DATA SET, EXTEND DATA SET, DELETE DATA SET, RESET DATA SET, and VSAM CATALOG ACCESS.

This value is an average.

Field Name: QWAXDSSE

EXC/DEL/DEF SUSP EVENTS

Number of wait trace events for waits for synchronous execution unit switching to the DB2 data space manager services.

Field Name: QWAXDSNS

OTHER SERVICE SUSP TIME

The VSAM catalog update. In the distributed environment, it includes the waiting time for the response from the server system.

Field Name: QWAXOTSE

OTHER SERVICE SUSP EVENTS

Number of wait trace events for a synchronous execution unit switch to other DB2 service tasks.

Field Name: QWAXOTNS

FORCE-AT-COMMIT SUSP TIME
IFCID 147 - Instrumentation Accounting Data Overflow

The accumulated time waiting for phase 1 commit write I/O. An example for this suspension is LOB Table Space with LOG NO Phase 1 commit database synchronous write I/O processing.

Field Name: QWAXAWFC

FORCE-AT-COMMIT SUSP EVENTS

The number of wait trace events for force-at-commit.

Field Name: QWAXFCCT

ASYNC. IXL REQ TIME

The accumulated wait time for IXLCACHE and IXLFCOMP requests.

Field Name: QWAXIXLT

ASYNC. IXL EVENTS

Number of wait trace events processed for asynchronous IXLCACHE or IXLFCOMP invocations.

Field Name: QWAXIXLE
IFCID 147 - Logging

This topic shows detailed information about “Record Trace - IFCID 147 - Logging”.

Record trace - IFCID 147 - Logging

The field labels shown in the following sample layout of “Record Trace - IFCID 147 - Logging” are described in the following section.

| LOGGING | NUMBER OF LOG RECORDS WRITTEN | 0 | TOTAL BYTES WRITTEN | 0 |

**NUMBER OF LOG RECORDS WRITTEN**

The number of log records written.

Field Name: QWACLRN

**TOTAL BYTES WRITTEN**

The number of log record bytes written. This field is calculated from DB2 fields QWACLRAB x QWACLRN.

Field Name: RT0147BW
IFCID 147 - Monitor Detail Data

This topic shows detailed information about “Record Trace - IFCID 147 - Monitor Detail Data”.

Record trace - IFCID 147 - Monitor Detail Data

The field labels shown in the following sample layout of “Record Trace - IFCID 147 - Monitor Detail Data” are described in the following section.

API BEGIN ELAPSED TIME

The API begin elapsed time in the format day, hour, minute, second, and millionth of a second. If the value is 0, N/P is printed.

Field Name: QW0148AB

API ENDING ELAPSED TIME

The API ending elapsed time in the format day, hour, minute, second, and millionth of a second. If the value is 0, N/P is printed.

Field Name: QW0148AE

API BEGIN CPU TIME

The API beginning CPU time in the format minute, second, and millionth of a second. If the value is 0, N/P is printed.

Field Name: QW0148UB

API ENDING CPU TIME

The API ending CPU time in the format minute, second, and millionth of a second. If the value is 0, N/P is printed.

Field Name: QW0148UE

API BEGIN SRB TIME

The API beginning SRB time in the format minute, second, and millionth of a second. If the value is 0, N/P is printed.

Field Name: QW0148SB

API ENDING SRB TIME
IFCID 147 - Monitor Detail Data

The API ending SRB time in the format minute, second, and millionth of a second. If the value is 0, N/P is printed.

Field Name: QW0148SE

IFI BEGIN ELAPSED TIME

The IFI begin elapsed time in the format day, hour, minute, second, and millionth of a second. If the value is 0, N/P is printed.

Field Name: QW148IAB

IFI ENDING ELAPSED TIME

The IFI ending elapsed time in the format day, hour, minute, second, and millionth of a second. If the value is 0, N/P is printed.

Field Name: QW148IAE

IFI BEGIN CPU TIME

The IFI beginning CPU time in the format minute, second, and millionth of a second. If the value is 0, N/P is printed.

Field Name: QW148IUB

IFI ENDING CPU TIME

The IFI ending CPU time in the format minute, second, and millionth of a second. If the value is 0, N/P is printed.

Field Name: QW148IUE

BEGIN/RESUME CPU TIME

The start or resume CPU time for a stored procedure, user-defined function, or trigger.

Field Name: QW148ATC

BEGIN/RESUME TOD TIME

The start or resume TOD time for a stored procedure, user-defined function, or trigger.

Field Name: QW148AOD

LOCK - I/O - LATCH BEGIN TIME

The beginning time of the I/O including lock and latch use in the format day, hour, minute, second, and millionth of a second. If the value is 0, N/P is printed.

Field Name: QW0148LB

LOCK - I/O - LATCH ENDING TIME

The ending time of the I/O including lock and latch use in the format day, hour, minute, second, and millionth of a second. If the value is 0, N/P is printed.

Field Name: QW0148LE

END-OF-TASK CPU TIME

The CPU time from DSN3EOT0 in the format minute, second, and millionth of a second. If the value is 0, N/P is printed.

Field Name: QW0148EO
ACCOUNTING ENTRY CPU TIME
The CPU time at entry to a monitoring or accounting class 2 or class 3 wait, in the format minute, second, and millionth of a second. If the value is 0, N/P is printed.
Field Name: QW0148LW

EU SWITCH BEGIN ELAPSED TIME
The beginning of the elapsed time of the wait for the execution unit switch. If the value is 0, N/P is printed.
Field Name: QW0148EB

EU SWITCH ENDING ELAPSED TIME
The end of the elapsed time of the wait for the execution unit switch. If the value is 0, N/P is printed.
Field Name: QW0148EE

ARCH LOG(QUIESCE) BEGIN TIME
The beginning of the elapsed time of the wait for the ARCHIVE LOG MODE (QUIESCE) command. If the value is 0, N/P is printed.
Field Name: QW0148RB

ARCH LOG(QUIESCE) ENDING TIME
The end of the elapsed time of the wait for the ARCHIVE LOG MODE (QUIESCE) command. The end time minus begin time should be the total time the agent is suspended due to the ARCHIVE LOG MODE (QUIESCE) command. If the value is 0, N/P is printed.
Field Name: QW0148RE

ACE TOKEN
The hexadecimal address of the agent control element. Indicates the thread reported here. If the value is 0, N/P is printed.
Field Name: QW0148AC

APPL REQUEST COUNT
The number of attachment facility calls to DB2.
Field Name: QW0148RQ

ASCB TOKEN
The ASCB token in hexadecimal. If the value is 0, N/P is printed.
Field Name: QW0148MA

LATCH TOKEN
The latch token. If the value is 0, N/P is printed.
Field Name: QW0148LA

TCB TOKEN
The TCB token. If the value is 0, N/P is printed.
Field Name: QW0148MT

AGENT ASID
IFCID 147 - Monitor Detail Data

The ASID of the thread.

Field Name: QW0148AS

STATUS INDICATOR 1
Indicates whether the thread is at plan or signon/identify level.

Field Name: QW0148CD

CONNECTION TYPE
The connection type.

Field Name: QW0148TY

LATEST IFCID
The latest IFCID processed.

Field Name: QW0148IL

STATUS INDICATOR 2
Indicates whether the agent is in end-of-task processing.

Field Name: QW0148ES

DBID
The database ID. Deduced from the DB2 fields QW0148DB, QW0105DN or QW0107DN.

When present, the database name is shown, otherwise the decimal identifier from QW0148DB is shown, or N/A when this value is 0.

Field Name: RT0148DB

PREVIOUS IFCID
The previous IFCID processed.

Field Name: QW0148IP

STATUS INDICATOR 3
Indicates whether the create thread request is queued.

Field Name: QW0148CQ

OBID
The object ID. Deduced from the DB2 fields QW0148OB, QW0105TN or QW0107TN.

If present, the name of the object is shown, otherwise the decimal identifier from QW0148OB is shown, or N/A if this value is 0.

Field Name: RT0148OB

LATCH CLASS
The latch class in hexadecimal.

Field Name: QW0148LC

TSO CONNECTION TYPE
The TSO connection type.

Field Name: QW0148TS

THREAD TYPE
IFCID 147 - Monitor Detail Data

The type of thread being processed.
Field Name: QW0148DD

THREAD STATUS
The status of the thread being processed.
Field Name: QW0148AI

STATUS INDICATOR 5
Indicates whether the agent is queued for end-of-task processing.
Field Name: QW0148EQ

THREAD TOKEN
The thread token. This token uniquely identifies a specific thread and also
appears in the display thread command response.
Field Name: QW148TTK

LUWSEQ
The LUW sequence number.
Field Name: QW0148CC

STATUS INDICATOR 6
Indicates whether the thread is running a stored procedure in DB2.
Field Name: QW0148SN

NESTING.LVL
Nesting level of the stored procedure, user-defined function or trigger, in the range 1 through 16.
Field Name: QW148ALV

STATUS INDICATOR 7
Indicates whether the thread is queued waiting for a stored procedure to be scheduled.
Field Name: QW0148SQ

NESTING ACTIVITY
Nesting activity of the stored procedure, user-defined function or trigger, if any.
Field Name: QW148AFG

CONS.TOKEN
The consistency token.
Field Name: QW0148CN

NETWORKID
The network identifier.
Field Name: QW0148NI

LUNAME
The logical unit name.
Field Name: QW0148LM
UNIQUENESS VALUE
The instance number.
Field Name: QW0148UV

ENCL.TOKEN
The enclave token, if under enclave, otherwise zero.
Field Name: QW148ETK

LOCATION
The name of the location where the thread executes the package.
Field Name: QW0148LN

COLLECTION
The collection name.
Field Name: QW0148CI

PACKAGE
The package identifier.
Field Name: QW0148PN

STORED PROCEDURE NAME
The stored procedure name.
Field Name: QW0148SP

SCHEMA NAME
Schema name, under which a stored procedure, user-define function or trigger is executing.
Field Name: QW148SCH

TOKEN
The consistency token.
Field Name: QW0148CN
IFCID 147 - Thread Correlation Data

IFCID 147 - Thread Correlation Data

This topic shows detailed information about “Record Trace - IFCID 147 - Thread Correlation Data”.

Record trace - IFCID 147 - Thread Correlation Data

The field labels shown in the following sample layout of “Record Trace - IFCID 147 - Thread Correlation Data” are described in the following section.

THREAD CORRELATION DATA

FIELD NAMES

ORIGINAL AUTHID: OPERID01
PRIMARY AUTHID: AUTHID01
PLAN NAME: PLANNAM1
CONNECTION NAME: CONNECT1
CONNECTION TYPE: DRDA PROTOCOL
CORRELATION ID: CORRELID0001
ACCOUNTING TOKEN (CHAR): ACCOUNTINGTOKEN0000001
ACCOUNTING TOKEN (HEX): x'C1C3C3D6E4D5E3C9D5C7E3D62C50F0F0F0F0F0F1'

ORIGINAL AUTHID

The original authorization ID. Possible values are:

- For TSO: the logon ID
- For batch: the user ID on the job statement
- For IMS (message-driven regions): the signon ID, LTERM, ASXBUSR, or PSB name
- For IMS (control regions): the user ID on the job statement, or the RACF started procedure entry if RACF is used
- For CICS: the user ID, TERM ID, TRAN ID, or as specified in the resource control table
- For MVS operator commands and DB2 system internal agents: SYSOPR
- For a distributed application server (AS):
  - If the application requester (AR) is a DB2 system, then this is the same value that was assigned at the AR.
  - If the application requester is not a DB2 system, then this is the user ID used to make the initial connection with the application server.

Field Name: QWHCOPID

PRIMARY AUTHID

The primary authorization ID from a connection or signon. The connection authorization exit and the signon authorization exit can change the primary authorization ID so that it differs from the original primary authorization ID (ORIGAUTH). Distributed authorization ID translation can also change the primary authorization ID.

Field Name: QWHCAID

CORRELATION ID

Correlation ID value:

FOR BATCH
  Job name

FOR TSO
  Logon ID

FOR IMS/VS
  PST#.PSBNAME

FOR CICS
  CONNECTION_TYPE.THREAD_TYPE.THREAD_.#.TRAN-ID

40-514  OMEGAMON XE for DB2 PE & PM: Report Reference
FOR RRSAF
   CORRELATION-ID VALUE FROM SIGNON FUNCTION

For threads using the DB2 private protocol or DRDA from a DB2 requester
   This field contains the correlation-id name of the thread at the requesting location.

For threads using DRDA from a non-DB2 requester
   This field contains the first 12 characters in the DDM external name (EXTNAM) parameter of the DDM EXCSAT command received as part of the SQL connect.

Field Name: QWHCCV

PLAN NAME
The plan name. It is blank for a DB2 command thread; otherwise:

DSNESPRR
   For SPUFI with repeatable read.

DSNESPCS
   For SPUFI with cursor stability.

DSNUTIL
   For utilities.

DSNTEP2
   For DSNTEP2.

DSNBIND
   For binding.

The application plan name
   For IMS.

The application plan name
   For CICS.

A blank plan name
   For IMS and CICS commands.

DSQPLAN
   For QMF.

The first 8 bytes of the application name
   For DRDA connections to the common servers.

Field Name: QWHCPLAN
   This is an exception field.

CONNECTION NAME
The connection name. Possible values are:
   • For batch: BATCH
   • For TSO: TSO
   • For QMF: DB2CALL
   • For utilities: UTILITY
   • For DB2 private protocol this is the DB2 subsystem ID
   • For IMS: the IMS ID
   • For CICS, this is the CICS ID
IFCID 147 - Thread Correlation Data

- For DRDA connections from non-DB2 requesters: SERVER
  
  **Field Name:** QWHCCN
  
  This is an *exception* field.

**CONNECTION TYPE**

The connecting system type code (in hexadecimal). This field can have a null value. Utilities, for example, do not have a connecting system type.

  **Field Name:** QWHCATYP
  
  This is an *exception* field.

**CORRELATION TOKEN**

Accounting correlation token.

This field applies only if CONNECTION TYPE equals CICS ATTACH or RRSAF ATTACH, otherwise N/A is shown.

If connection type is CICS ATTACH, the first eight bytes identify the network name (right padded with blanks), the second eight bytes identify the LU name (right padded with blanks), the final six bytes are the uniqueness value.

If the connection type is RRSAF ATTACH, the field is the value of the parameter accounting token in the RRSAF signon function.

This field is shown as both a character and a hexadecimal string.

  **Field Name:** QWHCTOKN
IFCID 149 - Resource Locking

This record only contains data from an Online Monitor trace data set.

Note: This report has the same layout as IFCID 150, for details refer to IFCID 150 - Thread Locking on page 40-518.
This record only contains data from an Online Monitor trace data set.

This topic shows detailed information about "Record Trace - IFCID 150 - Global Interest Data".

This topic shows detailed information about "Record Trace - IFCID 150 - Held Lock Data".

This topic shows detailed information about "Record Trace - IFCID 150 - Lock Resource Data".

This topic shows detailed information about "Record Trace - IFCID 150 - Retained Lock Data".

This topic shows detailed information about "Record Trace - IFCID 150 - Suspend Lock Data".
IFCID 150 - Global Interest Data

This topic shows detailed information about “Record Trace - IFCID 150 - Global Interest Data”.

Record trace - IFCID 150 - Global Interest Data

The field labels shown in the following sample layout of “Record Trace - IFCID 150 - Global Interest Data” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A PLOCK IS HELD</td>
<td>Indicates if a P-lock is held. It can have one of the following values:</td>
</tr>
<tr>
<td></td>
<td>• YES</td>
</tr>
<tr>
<td></td>
<td>• NO</td>
</tr>
<tr>
<td></td>
<td>• N/P</td>
</tr>
<tr>
<td>RESULTANT REQUESTED STATE</td>
<td>The result of the requested lock state. It can have one of the following</td>
</tr>
<tr>
<td></td>
<td>values:</td>
</tr>
<tr>
<td></td>
<td>• UNPROTECTED SHARE</td>
</tr>
<tr>
<td></td>
<td>• INTENTION SHARE</td>
</tr>
<tr>
<td></td>
<td>• INTENTION EXCLUSIVE</td>
</tr>
<tr>
<td></td>
<td>• SHARE</td>
</tr>
<tr>
<td></td>
<td>• UPDATE</td>
</tr>
<tr>
<td></td>
<td>• SHARE AND INTENTION EXCLUSIVE</td>
</tr>
<tr>
<td></td>
<td>• NON-SHARED UPDATE</td>
</tr>
<tr>
<td></td>
<td>• EXCLUSIVE</td>
</tr>
<tr>
<td></td>
<td>• INTENTION SHARE, LOWER LEVEL PLOCKS ACQUIRED</td>
</tr>
<tr>
<td></td>
<td>• INTENTION EXCLUSIVE, LOWER LEVEL PLOCKS ACQUIRED</td>
</tr>
<tr>
<td></td>
<td>• SHARED AND INTENTION EXCLUSIVE, LOWER LEVEL PLOCKS ACQUIRED</td>
</tr>
<tr>
<td>SUBSYSTEM NAME</td>
<td>The name of the subsystem.</td>
</tr>
<tr>
<td></td>
<td>Field Name: QW01505S</td>
</tr>
<tr>
<td>RESULTANT HELD STATE</td>
<td>The result of the requested P-lock held state. It can have one of the</td>
</tr>
<tr>
<td></td>
<td>following values:</td>
</tr>
<tr>
<td></td>
<td>• UNPROTECTED SHARE</td>
</tr>
<tr>
<td></td>
<td>• INTENTION SHARE</td>
</tr>
<tr>
<td></td>
<td>• INTENTION EXCLUSIVE</td>
</tr>
<tr>
<td></td>
<td>• SHARE</td>
</tr>
<tr>
<td></td>
<td>• UPDATE</td>
</tr>
<tr>
<td></td>
<td>• SHARE AND INTENTION EXCLUSIVE</td>
</tr>
</tbody>
</table>
IFCID 150 - Global Interest Data

- NON-SHARED UPDATE
- EXCLUSIVE
- INTENTION SHARE, LOWER LEVEL PLOCKS ACQUIRED
- INTENTION EXCLUSIVE, LOWER LEVEL PLOCKS ACQUIRED
- SHARED AND INTENTION EXCLUSIVE, LOWER LEVEL PLOCKS ACQUIRED

Field Name: QW01505H
IFCID 150 - Held Lock Data

This topic shows detailed information about “Record Trace - IFCID 150 - Held Lock Data”.

Record trace - IFCID 150 - Held Lock Data

The field labels shown in the following sample layout of “Record Trace - IFCID 150 - Held Lock Data” are described in the following section.

HELD LOCK DATA

LOCK REQUEST TOKEN : X'0C000002'
LOCK STATE : X'00'
SUBSYSTEM NAME : SYS00003
ACE TOKEN : X'00000BB9'
LOCK TYPE : P-LOCK
FUNCTION : LOCK
LOCK DURATION : X'00'
QW0150RW: X'0C000002'
QW0150UC: X'00000000'
QW0150SC: (1) 0 (2) 0 (3) 0 (4) 0 (5) 0 (6) 0 (7) 0 (8) 0

LOCK REQUEST TOKEN

The lock request token in hexadecimal.

Field Name: QW0150R3

LOCK STATE

The lock state.

Field Name: QW0150ST

SUBSYSTEM NAME

The name of the subsystem.

Field Name: QW0150N4

ACE TOKEN

The hexadecimal address of the agent control element indicating the holder of this lock.

Field Name: QW0150A3

LOCK TYPE

The type of lock.

Field Name: QW0150TL

FUNCTION

The lock function.

Field Name: QW0150F4

LOCK DURATION

The duration for which the lock is held:

MANUAL  Varies depending on the ISOLATION parameter
MANUAL+1  Temporary change of consistency level from CS to RR during bind and DDL
COMMIT  Until commit
COMMIT+1  Past commit; applies to locks needed to maintain the position for a cursor opened WITH HOLD
ALLOCATION  Until deallocation
IFCID 150 - Held Lock Data

PLAN  For the duration of the plan
FREE ALL LOCKS
    Until all locks are freed
Field Name: QW0150D4
IFCID 150 - Lock Resource Data

This topic shows detailed information about “Record Trace - IFCID 150 - Lock Resource Data”.

Record trace - IFCID 150 - Lock Resource Data

The field labels shown in the following sample layout of “Record Trace - IFCID 150 - Lock Resource Data” are described in the following section.

LOCK RES TYPE

The locked resource type.

Note: For data sharing, SKELETON CURSOR TABLE LOCKING and SKELETON PACKAGE TABLE LOCK are LP-locks (an LP-lock has an L-lock component and a P-lock component).

Field Name: QW0150KT

DBID

The database ID. This field is not applicable if the value in LOCK RES TYPE is SKELETON CURSOR TABLE LOCKING, SKELETON PACKAGE TABLE LOCK, COLLECTION, or ALTER BUFFER POOL.

Deduced from the DB2 fields QW0150DB, QW0105DN, or QW0107DN.

If present, the database name is shown, otherwise the decimal identifier from QW0150DB is shown, or N/A if this value is 0.

Field Name: RT0150DB

OBID

The object ID of the table space or pageset involved in the lock. This field is not applicable if the value in LOCK RES TYPE is SKELETON CURSOR TABLE LOCKING, SKELETON PACKAGE TABLE LOCK, or COLLECTION.

Deduced from the DB2 fields QW0149KP, QW0105TN, QW0107TN, QW0149KP, QW0105OB or QW0107OB.

If present, then name of the object is shown. Otherwise the decimal identifier from QW0150KP is shown, or N/A if this value is 0.

Field Name: RT0150OB

RESOURCE ID

The hexadecimal identifier of the small resource. If LOCK RES TYPE is:

DATA SET LOCKING (PARTITION)

Last byte is the partition number

DATA PAGE LOCKING

First 3 bytes are the page number

INDEX PAGE LOCKING

First 3 bytes are the page number and the last byte is the subpage number

HASH ANCHOR LOCK

First 3 bytes are the page number and the last byte is the anchor point ID
IFCID 150 - Lock Resource Data

**CS-READ DRAIN**
Last byte is the partition number (optional)

**RR-READ DRAIN**
Last byte is the partition number (optional)

**WRITE DRAIN**
Last byte is the partition number (optional)

**ROW LOCK**
First 3 bytes are the page number and the last byte is the row ID of the record

**INDEX END OF FILE LOCK**
Last byte is the partition number (optional)

**Note:** In large partitioned table spaces, the page number covers 4 bytes instead of 3.

This field is not applicable if the value in LOCK RES TYPE is SKELETON CURSOR TABLE LOCKING, SKELETON PACKAGE TABLE LOCK, COLLECTION, ALTER BUFFER POOL, or PAGESET LOCK. If the value is UTILITY SERIALIZATION LOCK or BINDLOCK, N/A is printed.

**Field Name:** QW0150KR

**HASH TOKEN**
The hash token of the resource name.

**Field Name:** QW0150LH
IFCID 150 - Retained Lock Data

This topic shows detailed information about “Record Trace - IFCID 150 - Retained Lock Data”.

Record trace - IFCID 150 - Retained Lock Data

The field labels shown in the following sample layout of “Record Trace - IFCID 150 - Retained Lock Data” are described in the following section.

LOCK REQUEST TOKEN
The lock request token in hexadecimal.
Field Name: QW0150T4

LOCK STATE
The lock state in hexadecimal.
Field Name: QW0150R4

SUBSYSTEM NAME
The name of the subsystem.
Field Name: QW0150N4
IFCID 150 - Suspend Lock Data

IFCID 150 - Suspend Lock Data
This topic shows detailed information about “Record Trace - IFCID 150 - Suspend Lock Data”.

Record trace - IFCID 150 - Suspend Lock Data
The field labels shown in the following sample layout of “Record Trace - IFCID 150 - Suspend Lock Data” are described in the following section.

LOCK REQUEST TOKEN
The lock request token in hexadecimal.
Field Name: QW0150R3

LOCK STATE
The lock state.
Field Name: QW0150ST

SUBSYSTEM NAME
The name of the subsystem.
Field Name: QW0150N4

ACE TOKEN
The hexadecimal address of the agent control element indicating the holder of this lock.
Field Name: QW0150A3

LOCK TYPE
The type of lock.
Field Name: QW0150TL

FUNCTION
The lock function.
Field Name: QW0150F4

LOCK DURATION
The duration for which the lock is held:
MANUAL Varies depending on the ISOLATION parameter
MANUAL+1 Temporary change of consistency level from CS to RR during bind and DDL
COMMIT Until commit
COMMIT+1 Past commit; applies to locks needed to maintain the position for a cursor opened WITH HOLD
ALLOCATION Until deallocation
**IFCID 150 - Suspend Lock Data**

**PLAN**  For the duration of the plan

**FREE ALL LOCKS**  Until all locks are freed

**Field Name:** QW0150D4
IFCID 151 - User Record

This topic shows detailed information about “Record Trace - IFCID 151 - User Record”.

When present, data is shown in hexadecimal dump format.
IFCID 152 - User Record

This topic shows detailed information about “Record Trace - IFCID 152 - User Record”.

When present, data is shown in hexadecimal dump format.
IFCID 153 - User Record

This topic shows detailed information about “Record Trace - IFCID 153 - User Record”.

When present, data is shown in hexadecimal dump format.
IFCID 154 - User Record

This topic shows detailed information about “Record Trace - IFCID 154 - User Record”.

When present, data is shown in hexadecimal dump format.
IFCID 155 - User Record

This topic shows detailed information about “Record Trace - IFCID 155 - User Record”.

When present, data is shown in hexadecimal dump format.
This topic shows detailed information about “Record Trace - IFCID 156 - User Record”.

When present, data is shown in hexadecimal dump format.
IFCID 157 - DRDS RDS Interface

Record trace - IFCID 157 - DRDS RDS Interface

The field labels shown in the following sample layout of “Record Trace - IFCID 157 - DRDS RDS Interface” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EVENT</td>
<td>The type of event. Field Name: QW0157E</td>
</tr>
<tr>
<td>REQUEST</td>
<td>The type of request. This field shows N/P if the value in EVENT is RETURN. Field Name: QW0157O</td>
</tr>
<tr>
<td>PGM</td>
<td>The program name. This field shows N/P if the value in EVENT is RETURN. Field Name: QW0157PN</td>
</tr>
<tr>
<td>CALL</td>
<td>The type of call. This field shows N/P if the value in EVENT is RETURN. Field Name: QW0157CT</td>
</tr>
<tr>
<td>PLAN SECTN</td>
<td>The section number in the plan. This field shows N/P if the value in EVENT is RETURN. Field Name: QW0157SN</td>
</tr>
<tr>
<td>SERVING LOC</td>
<td>The name of the server location. This field shows N/P if the value in EVENT is RETURN. Field Name: QW0157LN</td>
</tr>
</tbody>
</table>
IFCID 158 - DRDS CNV Interface

This topic shows detailed information about “Record Trace - IFCID 158 - DRDS CNV Interface”.

Record trace - IFCID 158 - DRDS CNV Interface

The field labels shown in the following sample layout of “Record Trace - IFCID 158 - DRDS CNV Interface” are described in the following section.

DRDS CNV 'BLANK'
INTERFACE NETWORKID: DEIBMIPS LUNAME: IPSAR721 LUWSEQ: 1
EVENT: RETURN CALL TYPE : N/P
PROGRAM : N/P

EVENT

The type of event.

Field Name: QW0158E

CALL TYPE

The type of call. This field shows N/P if the value in EVENT is RETURN.

Field Name: QW0158CT

PGM

The name of the program. This field shows N/P if the value in EVENT is RETURN.

Field Name: QW0158PN

PLAN SECTN

The section number within the plan. This field shows N/P if the value in EVENT is RETURN.

Field Name: QW0158SN
IFCID 159 - DRDS Req Site Data

Record trace - IFCID 159 - DRDS Req Site Data

The field labels shown in the following sample layout of “Record Trace - IFCID 159 - DRDS Req Site Data” are described in the following section.

DRDS REQ 'BLANK'
SITE DATA   NETWORKID: DEIBMIPS LUNAME: IPSAR721 LUWSEQ: 1
EVENT : WAIT RESP
SERVLOC: N/P
CONVID : X'00000002'   GPR15 :  8

EVENT
The type of event.
Field Name: QW0159E

SERVLOC
The name of the server location. This field shows N/P if the value in EVENT is WAIT RESP.
Field Name: QW0159LN

CONVID
The conversation identification number.
Field Name: QW0159CI

GPR15
The return code in general purpose register 15. This field shows N/P if the value in EVENT is CREATE CONV.
Field Name: QW015915
IFCID 160 - DC Requester

This topic shows detailed information about “Record Trace - IFCID 160 - DC Requester”.

Record trace - IFCID 160 - DC Requester

The field labels shown in the following sample layout of “Record Trace - IFCID 160 - DC Requester” are described in the following section.

```
EVENT: ALLOCATE CONVERSATION
MSGTYPE: N/P
MSG RESPONSE: N/P
MSG LGTH: N/P
QW0160ID X'21'
QW0160CI X'00BF3128'
QW0160VI X'01000003'
QW0160SI X'0073F4923DC3D965'
QW0160LM X'E2E8E2C1C4D4D3D4'
QW0160VT X'4D000000060E2CED'
QW0160DA X'0000000000000000'
```

**EVENT**

The type of event.

*Field Name:* QW0160E

**MSGTYPE**

The message type. This field is only applicable if the value in EVENT is RECEIVE RESPONSE MESSAGE or SEND REQUEST MESSAGE.

*Field Name:* QW0160T

**MSG RESPONSE**

The message response. This field is only applicable if the value in EVENT is RECEIVE RESPONSE MESSAGE or SEND REQUEST MESSAGE.

*Field Name:* QW0160R

**MSG LGTH**

The message length. This field is only applicable if the value in EVENT is RESET CONVERSATION, RECEIVE RESPONSE MESSAGE, or SEND REQUEST MESSAGE.

*Field Name:* QW0160ML

**MSGCLASS**

The message class. This field is only applicable if the value in EVENT is RESET CONVERSATION, RECEIVE RESPONSE MESSAGE, SEND REQUEST MESSAGE, or WAIT FOR RESPONSE MESSAGE.

*Field Name:* QW0160MC

**MSGNO**

The message number. This field is only applicable if the value in EVENT is RESET CONVERSATION, RECEIVE RESPONSE MESSAGE, SEND REQUEST MESSAGE, or WAIT FOR RESPONSE MESSAGE.

*Field Name:* QW0160MN

**MSGTIME**

The timestamp at the start of the VTAM request.

*Field Name:* QW0160MS
**Record trace - IFCID 161 - DC Server**

The field labels shown in the following sample layout of “Record Trace - IFCID 161 - DC Server” are described in the following section.

<table>
<thead>
<tr>
<th>EVENT</th>
<th>MSGTYPE</th>
<th>MSG RESPONSE</th>
<th>MSG LGTH</th>
<th>MSGCLASS</th>
<th>MSGNO</th>
<th>MSGTIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>RECEIVE REQUEST MESSAGE</td>
<td>REQUEST</td>
<td>DATA</td>
<td>756</td>
<td>4</td>
<td>0</td>
<td>03/13/08 23:18:23.315984</td>
</tr>
<tr>
<td>QW0161ID</td>
<td>X'11'</td>
<td>QW0161CI</td>
<td>X'1032128'</td>
<td>QW0161VI</td>
<td>X'01000008'</td>
<td>QW0161SI</td>
</tr>
<tr>
<td>QW0161LM</td>
<td>X'C9C2D4D9C4C24040'</td>
<td>QW0161VT</td>
<td>X'0000000000000000'</td>
<td>QW0161DA</td>
<td>X'0000000000000000'</td>
<td></td>
</tr>
</tbody>
</table>

**EVENT**

The type of event.

*Field Name:* QW0161E

**MSGTYPE**

The message type. This field is only applicable if the value in EVENT is RECEIVE REQUEST MESSAGE or SEND RESPONSE MESSAGE.

*Field Name:* QW0161T

**MSG RESPONSE**

The message response. This field is only applicable if the value in EVENT is RECEIVE REQUEST MESSAGE or SEND RESPONSE MESSAGE.

*Field Name:* QW0161R

**MSG LGTH**

The message length. This field is only applicable if the value in EVENT is RECEIVE REQUEST MESSAGE or SEND RESPONSE MESSAGE.

*Field Name:* QW0161ML

**MSGCLASS**

The message class. This field is only applicable if the value in EVENT is RECEIVE REQUEST MESSAGE or SEND RESPONSE MESSAGE.

*Field Name:* QW0161MC

**MSGNO**

The message number. This field is only applicable if the value in EVENT is RECEIVE REQUEST MESSAGE or SEND RESPONSE MESSAGE.

*Field Name:* QW0161MN

**MSGTIME**

Message timestamp. This field is only applicable if the value in EVENT is RECEIVE REQUEST MESSAGE or SEND RESPONSE MESSAGE.

*Field Name:* QW0161MS
This topic shows detailed information about “Record Trace - IFCID 162 - DTM Request”.

**Record trace - IFCID 162 - DTM Request**

The field labels shown in the following sample layout of “Record Trace - IFCID 162 - DTM Request” are described in the following section.

DTM REQUEST  NETWORKID: G91E81D0  LUNAME: D179  LUWSEQ: 6  
REQUESTING LOCATION: 9.30.129.208  
REQUESTING TIMESTAMP: N/P  
AR NAME: gixxer  PRDID: CLNT/SER V8 R1 M4  
ACCTKN X'c7f9f1c5f8f1c4f04bc4f1f7f7f9000f92022652404040'  
EVENT: DEALLOCATION INITIATED  
LOCATION TYPE: N/P

**EVENT**

The type of event.

**Field Name:** QW0162E

**LOCATION TYPE**

The type of location.

**Field Name:** QW0162LT

**LOCATION NAME**

The name of the DB2 location where this event occurred.

**Field Name:** QW0162LN
This topic shows detailed information about “Record Trace - IFCID 163 - DTM Respond”.

Record trace - IFCID 163 - DTM Respond

The field labels shown in the following sample layout of “Record Trace - IFCID 163 - DTM Respond” are described in the following section.

EVENT: DBAT CREATED AT SERVER

EVENT

The event type.

Field Name: QW0163E
IFCID 164 - IBM Service Record

This topic shows detailed information about “Record Trace - IFCID 164 - IBM Service Record”.

This record is for IBM service use.
IFCID 165 - IBM Service Record

This topic shows detailed information about “Record Trace - IFCID 165 - IBM Service Record”.

This record is for IBM service use.

Record trace - IFCID 165 - IBM Service Record

The field labels shown in the following sample layout of “Record Trace - IFCID 165 - IBM Service Record” are described in the following section.

- **QW0165MN**: X’1122334455667788’
- **QW0165RC**: 64
- **QW0165MC**: X’C7C5E3C1C4C9D5C6’
- **QW0165FO**: 32
- **QW0165MQ**: X’1234567890ABCDEF’
- **QW0165RP**: 16
- **QW0165CI**: N/A
- **QW0165RS**: 8
- **QW0165IPV6**: X’404040C7D9C1D5E34040404040404040’

**QW0165MN**

This field is for IBM service only.
- **Field Name**: QW0165MN

**QW0165RC**

This field is for IBM service only.
- **Field Name**: QW0165RC

**QW0165MC**

This field is for IBM service only.
- **Field Name**: QW0165MC

**QW0165FO**

This field is for IBM service only.
- **Field Name**: QW0165FO

**QW0165MQ**

This field is for IBM service only.
- **Field Name**: QW0165MQ

**QW0165RP**

This field is for IBM service only.
- **Field Name**: QW0165RP

**QW0165CI**

This field is for IBM service only.
- **Field Name**: QW0165CI

**QW0165RS**

This field is for IBM service only.
- **Field Name**: QW0165RS

**QW0165IPV6**

This field is for IBM service only.
- **Field Name**: QW0165IPV6
IFCID 166 - IBM Service Record

This topic shows detailed information about “Record Trace - IFCID 166 - IBM Service Record”.

This record is for IBM service use.
This topic shows detailed information about “Record Trace - IFCID 167 - Conv Alloc Req Queued”.

Record trace - IFCID 167 - Conv Alloc Req Queued

The field labels shown in the following sample layout of “Record Trace - IFCID 167 - Conv Alloc Req Queued” are described in the following section.

<table>
<thead>
<tr>
<th>Field</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONVID</td>
<td>QW0167CI</td>
</tr>
<tr>
<td>LU NAME</td>
<td>QW0167LU</td>
</tr>
<tr>
<td>CONV ALLOC</td>
<td>QW0167CA</td>
</tr>
<tr>
<td>MODE NAME</td>
<td>QW0167MO</td>
</tr>
<tr>
<td>CONV QUEUED</td>
<td>QW0167CQ</td>
</tr>
<tr>
<td>CONV LIMIT</td>
<td>QW0167CL</td>
</tr>
</tbody>
</table>

CONVID
The conversation identifier.

Field Name: QW0167CI

LU NAME
The logical unit name.

Field Name: QW0167LU

CONV ALLOC
The conversation allocated.

Field Name: QW0167CA

MODE NAME
The mode name.

Field Name: QW0167MO

CONV QUEUED
The conversation queued.

Field Name: QW0167CQ

CONV LIMIT
The conversation limit.

Field Name: QW0167CL
This topic shows detailed information about “Record Trace - IFCID 168 - IBM Service Record”.

This record is for IBM service use.
IFCID 169 - DIST Authid Translation

This topic shows detailed information about “Record Trace - IFCID 169 - DIST Authid Translation”.

Record trace - IFCID 169 - DIST Authid Translation

The field labels shown in the following sample layout of “Record Trace - IFCID 169 - DIST Authid Translation” are described in the following section.

IDENTIFIER TYPE: TRUSTED CONTEXT SYSTEM AUTHID

TRANSL TYPE: INBOUND

RESPOND LINKNAME: RESPLNK1

RESPOND LOC: RESPLOCATION0001

SYST AUTHID: REQAUTH1

TRANS AUTHID: TRLAUTH1

IDENTIFIER TYPE

Identifier type of the source ID. Possible values are:

A AUTHID

L LOCATION ALIAS

D DATABASE ALIAS

S TRUSTED CONTEXT SYSTEM AUTHID

Field Name: QW0169ID

TRANSL TYPE

The type of translation:

INBOUND

The responding DB2 site translates the AUTHID after receiving the data.

OUTBOUND

The requesting DB2 site translates the AUTHID before sending the data.

Field Name: QW0169TY

RESPOND LINKNAME

The logical unit name.

If the value of TRANSITION TYPE is INBOUND, this is the VTAM LU name of the requester location or row in SYSIBM.SYSIPNAMES.

If the value of TRANSITION TYPE is OUTBOUND, this is the VTAM LU name of the remote server or

Field Name: QW0169LU

RESPOND LOC

If the value of TRANSITION TYPE is INBOUND, this is the service location name regardless of whether the server is another DB2. If the value of TRANSITION TYPE is OUTBOUND, this field contains one of the following values:

- The name of the requesting DB2 location
- <LUNAME>
- NNN.NNN.NNN.NNN
IFCID 169 - DIST Authid Translation

Field Name: QW0169LO

SYST AUTHID

Either authorization ID or location or alias before translation. Depending on the translation type OUTBOUND and identifier type TRUSTED CONTEXT, this field contains the original value of the authorization ID or the system authorization ID.

Field Name: QW0169AU

TRANS AUTHID

The new value of the authorization ID, location, or alias. Depending on the translation type OUTBOUND and identifier type TRUSTED CONTEXT, this field contains the value of the authorization ID or the translated system authorization ID.

Field Name: QW0169NE
IFCID 170 - Suspend of Agent

IFCID 170 - Suspend of Agent

This topic shows detailed information about “Record Trace - IFCID 170 - Suspend of Agent”.

Record trace - IFCID 170 - Suspend of Agent

The field labels shown in the following sample layout of “Record Trace - IFCID 170 - Suspend of Agent” are described in the following section.

ACE: 1
QW0170ID X'01' QW0170FC X'05'

ACE

Indicates the requester. The relative number of the agent control element address in the ACE cross-reference table. This table is printed at the end of each location for every trace specified.

Field Name: QW0170AC
This topic shows detailed information about “Record Trace - IFCID 171 - IBM Service Record”.

This record is for IBM service use.
IFCID 172 - Deadlock Data

This topic shows the records available for deadlock data.

"IFCID 172 - Deadlock Header" on page 40-551
This topic shows detailed information about "Record Trace - IFCID 172 - Deadlock Header".

"IFCID 172 - Unit of Work - Resource" on page 40-552
This topic shows detailed information about "Record Trace - IFCID 172 - Unit of Work - Resource".

"IFCID 172 - Unit of Work - Blocker" on page 40-553
This topic shows detailed information about "Record Trace - IFCID 172 - Unit of Work - Blocker".

"IFCID 172 - Unit of Work - Waiter" on page 40-556
This topic shows detailed information about "Record Trace - IFCID 172 - Unit of Work - Waiter".
IFCID 172 - Deadlock Header

This topic shows detailed information about “Record Trace - IFCID 172 - Deadlock Header”.

This record is written every time that DB2 takes action to resolve a deadlock. This record details all of the units of work involved in the deadlock, the resource for which they were contending, and the attributes of their requests. One record is written for each locked resource in the deadlock.

Statistics class 3 or performance class 6 trace must be active.

DB2 can resolve a deadlock either by rolling back a unit of work for one of the agents involved, or by requesting a process to terminate.

There is no correlation between the number of IFCID 172 records written and the number of deadlocks counted by IFCIDs 2 and 3. Deadlocks can be broken without intervention by DB2, as an example when a process times out. Also, multiple IFCID 172 records can be written for a single deadlock.

Record trace - IFCID 172 - Deadlock Header

The field labels shown in the following sample layout of “Record Trace - IFCID 172 - Deadlock Header” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>INTERVAL COUNT</td>
<td>The deadlock interval counter. Field Name: QW0172IT</td>
</tr>
<tr>
<td>WAITERS INVOLVED</td>
<td>The number of waiters involved in the deadlock. Field Name: QW0172NR</td>
</tr>
<tr>
<td>TIME DETECTED</td>
<td>The date and time when the deadlock was detected. Field Name: QW0172TD</td>
</tr>
</tbody>
</table>
IFCID 172 - Unit of Work - Resource

IFCID 172 - Unit of Work - Resource
This topic shows detailed information about “Record Trace - IFCID 172 - Unit of Work - Resource”.

The content of the LOCK RES TYPE field determines which other fields are printed in this record.

Blocker and waiter information is shown for each resource involved in the deadlock.

Record trace - IFCID 172 - Unit of Work - Resource

The field labels shown in the following sample layout of “Record Trace - IFCID 172 - Unit of Work - Resource” are described in the following section.

UNIT OF WORK
RESOURCE
LOCK RES TYPE: LPL RECOVERY
DBID: 1
OBID: 2
LOCK HASH VALUE: X’005859E8’

LOCK RES TYPE
The locked resource type.

Note: For data sharing, SKELETON CURSOR TABLE LOCKING and SKELETON PACKAGE TABLE LOCK are LP-locks (an LP-lock has an L-lock component and a P-lock component).

Field Name: QW0172FR

DBID
The database ID. This field is not applicable if the value in LOCK RES TYPE is SKELETON CURSOR TABLE LOCKING, SKELETON PACKAGE TABLE LOCK, or COLLECTION. Deduced from the DB2 fields QW0172DB, QW0105DN, or QW0107DN.

If present, the database name is shown, otherwise the decimal identifier from QW0172DB is shown, or N/A if this value is 0.

Field Name: RT0172DB

OBID
The object ID of the resource involved in the lock. This field is not applicable if the value in LOCK RES TYPE is SKELETON CURSOR TABLE LOCKING, SKELETON PACKAGE TABLE LOCK, or COLLECTION.

Deduced from the DB2 fields QW0172KP, QW0105TN, QW0107TN, QW0172KP, QW0105OB, or QW0107OB.

If present, then name of the object is shown. Otherwise the decimal identifier from QW0172KP is shown, or N/A if this value is 0.

Field Name: RT0172OB

LOCK HASH VALUE
The hash value of the locked resource.

Field Name: QW0172LH
IFCID 172 - Unit of Work - Blocker

This topic shows detailed information about “Record Trace - IFCID 172 - Unit of Work - Blocker”.

A blocker is a thread that prevents its victim from acquiring its lock. The blocker might be a holder of the lock, or it might be another waiter (that came in before the victim) that is incompatible with the holder’s lock.

Note: If the fields PLAN NAME, CORR ID, CONN, and NETWORKID show an asterisk (*), the blocking request was released by the requester or was timed out between the detection and reporting of the deadlock.

Record trace - IFCID 172 - Unit of Work - Blocker

The field labels shown in the following sample layout of “Record Trace - IFCID 172 - Unit of Work - Blocker” are described in the following section.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRIMAUTH</td>
<td>The authorization ID of the thread holding the resource. Field Name: QW0172HB</td>
</tr>
<tr>
<td>PLAN NAME</td>
<td>The plan name of the blocker. Field Name: QW0172HP</td>
</tr>
<tr>
<td>CORR ID</td>
<td>The correlation name of the blocker. Field Name: QW0172HR</td>
</tr>
<tr>
<td>CONN ID</td>
<td>The connection ID of the blocker. Field Name: QW0172HN</td>
</tr>
<tr>
<td>NETWORKID</td>
<td>Provides the following information:</td>
</tr>
<tr>
<td></td>
<td>• The logical unit of work identifier of the blocker. The data shown is only</td>
</tr>
<tr>
<td></td>
<td>valid for distributed threads.</td>
</tr>
<tr>
<td></td>
<td>• The logical unit name of the blocker. The data shown is only valid for</td>
</tr>
<tr>
<td></td>
<td>distributed threads.</td>
</tr>
<tr>
<td></td>
<td>• The uniqueness value of the blocker. The data shown is only valid for</td>
</tr>
<tr>
<td></td>
<td>distributed threads.</td>
</tr>
<tr>
<td>LUNAME</td>
<td></td>
</tr>
</tbody>
</table>

Prior to DB2 10:
- CONS TOKEN: X'C1C2C3C4C5C6C7C8'
- STMT ID: X'000A0000'
- TYPE: N/A

DB2 10 or later:
- CONS TOKEN: X'C1C2C3C4C5C6C7C8'
- STMT ID: X'1234567890123456'
- TYPE: X'0000'
- STATUS: HOLD
- QW0172HF: X'12'
IFCID 172 - Unit of Work - Blocker

Provides the following information:
- The logical unit of work identifier of the blocker. The data shown is only valid for distributed threads.
- The logical unit name of the blocker. The data shown is only valid for distributed threads.
- The uniqueness value of the blocker. The data shown is only valid for distributed threads.

Field Name: QW0172HL

OWNING WORK UNIT
The owning unit of work of the blocker.
Field Name: QW0172HO

UNIQUENESS VALUE
Provides the following information:
- The logical unit of work identifier of the blocker. The data shown is only valid for distributed threads.
- The logical unit name of the blocker. The data shown is only valid for distributed threads.
- The uniqueness value of the blocker. The data shown is only valid for distributed threads.

Field Name: QW0172HL

MEMBER
The DB2 member name.
Field Name: QW0172HI

DURATION
The lock duration of the thread blocking the resource.
Field Name: QW0172HD

STATE
The lock state of the thread holding the resource.
Field Name: QW0172HS

ACE
The owning unit of work of the blocker.
Field Name: QW0172HO

TRANSACTION
The transaction or application name that is run.
Field Name: QWHCEUTX

WS_NAME
The end user's workstation name.
Field Name: QWHCEUWN

END_USER
IFCID 172 - Unit of Work - Blocker

The user ID of the workstation end user. This user ID can be different from the authorization ID used to connect to DB2. This field contains blanks if the client does not supply this information.

Field Name: QWHCEUID

PROGRAM NAME

The name of the program that is in control at the time of the deadlock. It need not be the program that acquired the lock.

Field Name: QW0172Q1

LOCATION

Location of the program that is in control at the time of the deadlock. It need not be the program that acquired the lock.

Field Name: QW0172Q3

PCKG/COLL ID

Package collection ID of the program that is in control at the time of the deadlock. It need not be the program that acquired the lock.

Field Name: QW0172Q2

CONS TOKEN

Consistency token of the program that is in control at the time of the deadlock. It need not be the program that acquired the lock.

Field Name: QW0172Q4

STMT ID

Shows the cached statement ID for the thread holding the resource. This field contains zero (0) if the client does not supply this information.

Field Name: QW0172H9

STMT ID

The holder statement ID.

Field Name: QW0172HZ

TYPE

The holder statement information.

Field Name: QW0172HY

STATUS

The status of the blocker.

WAIT The blocker is waiting for the resource.

HOLD The blocker is holding the resource.

Field Name: QW0172H2
IFCID 172 - Unit of Work - Waiter

IFCID 172 - Unit of Work - Waiter

This topic shows detailed information about “Record Trace - IFCID 172 - Unit of Work - Waiter”.

Record trace - IFCID 172 - Unit of Work - Waiter

The field labels shown in the following sample layout of “Record Trace - IFCID 172 - Unit of Work - Waiter” are described in the following section.

PRIMAUTH

The authorization ID of the thread waiting for the resource.

Field Name: QW0172WB

PLAN NAME

The plan name of the waiter.

Field Name: QW0172WP

CORR ID

The correlation ID of the waiter.

Field Name: QW0172WR

CONN ID

The connection ID of the waiter.

Field Name: QW0172WN

NETWORKID

Provides the following information:

- The logical unit of work identifier of the waiter. The data shown is only valid for distributed threads.
- The logical unit name of the waiter. The data shown is only valid for distributed threads.
- The uniqueness value of the waiter. The data shown is only valid for distributed threads.

Field Name: QW0172WL

LUNAME

Provides the following information:

- The logical unit of work identifier of the waiter. The data shown is only valid for distributed threads.
- The logical unit name of the waiter. The data shown is only valid for distributed threads.
- The uniqueness value of the waiter. The data shown is only valid for distributed threads.

Field Name: QW0172WL
OWNING WORK UNIT
The owning unit of work of the waiter.
Field Name: QW0172WO

UNIQUENESS VALUE
Provides the following information:
• The logical unit of work identifier of the waiter. The data shown is only valid for distributed threads.
• The logical unit name of the waiter. The data shown is only valid for distributed threads.
• The uniqueness value of the waiter. The data shown is only valid for distributed threads.
Field Name: QW0172WL

MEMBER
The waiter's DB2 member name.
Field Name: QW0172WI

DURATION
The lock duration of the thread waiting for the resource.
MANUAL Varies depending on the ISOLATION parameter (QW0172DR=x'20')
MANUAL+1 Temporary change of consistency level from CS to RR during bind and DDL (QW0172DR=x'21')
COMMIT Until commit (QW0172DR=x'40')
COMMIT+1 Past commit; applies to locks needed to maintain the position for a cursor opened WITH HOLD (QW0172DR=x'41')
ALLOCATION Until deallocation (QW0172DR=x'60')
PLAN For the duration of the plan (QW0172DR=x'80')
UTIL For the duration of the utility execution (QW0172DR=x'81')
INTEREST Duration used for P-locks (QW0172DR=x'FE')
FREE ALL Until all locks are freed (QW0172DR=x'FF')
N/A Not applicable for NOTIFY SUSPEND
Field Name: QW0172WD

STATE
The lock state of the thread waiting for the resource.
Field Name: QW0172WS

ACE
The owning unit of work of the waiter.
Field Name: QW0172WO
IFCID 172 - Unit of Work - Waiter

**TRANSACTION**
The transaction or application name that is run.

*Field Name:* QWHCEUTX

**WS_NAME**
The end user's workstation name.

*Field Name:* QWHCEUWN

**END_USER**
The user ID of the workstation end user. This user ID can be different from the authorization ID used to connect to DB2. This field contains blanks if the client does not supply this information.

*Field Name:* QWHCEUID

**PROGRAM NAME**
The name of the program that is in control at the time of the deadlock. It need not be the program that acquired the lock.

*Field Name:* QW0172Q5

**LOCATION**
Package collection ID of the program that is in control at the time of the deadlock. It need not be the program that acquired the lock.

*Field Name:* QW0172Q6

**PCKG/COLL ID**
Location of the program that is in control at the time of the deadlock. It need not be the program that acquired the lock.

*Field Name:* QW0172Q7

**CONS TOKEN**
Consistency token of the program that is in control at the time of the deadlock. It need not be the program that acquired the lock.

*Field Name:* QW0172Q8

**STMT ID**
Shows the cached statement ID for the thread waiting for the resource. This field contains zero (0) if the client does not supply this information.

*Field Name:* QW0172W9

**STMT ID**
The waiter statement ID.

*Field Name:* QW0172WZ

**TYPE**
The waiter statement information.

*Field Name:* QW0172WY

**DB2S ASIC**
The DB2S ASIC of the waiter. A unique number allocated to the requesting work unit of the waiter. The EB pointer of the waiter.
Field Name: QW0172AS

REQ WORK UNIT
The waiter's requesting work unit.
Field Name: QW0172UW

EB PTR
The EB pointer of the waiter.
Field Name: QW0172WE

REQ FUNCTION
The function requested by the waiter.
Field Name: QW0172WF

WORTH
The worth value DB2 assigns to the waiter.
Field Name: QW0172WA
IFCID 173 - Class 2 Time

This topic shows detailed information about “Record Trace - IFCID 173 - Class 2 Time”.

When present, data for this IFCID is printed in dump format, otherwise NO DATA is printed.

Record trace - IFCID 173 - Class 2 Time

The field labels shown in the following sample layout of “Record Trace - IFCID 173 - Class 2 Time” are described in the following section.

AUTH ID
The authorization ID.
Field Name: QW0173ID

PACKAGE COLL ID
The package collection ID.
Field Name: QW0173PC

PACKAGE NAME
The package name.
Field Name: QW0173PK

CURSOR NAME
The cursor name, if there is a cursor.
Field Name: QW0173CN

PLAN NAME
The plan name.
Field Name: QW0173PL

SECTION NUMBER
The section number in the plan.
Field Name: QW0173SN

STATEMENT NUMBER
The statement number in the plan.
Field Name: QW0173ST

CACHED STMNT ID
The cached statement ID. Zero (0) indicates that this information is not supplied.
Field Name: QW0173CS
IFCID 174 - Arch Log CMD Sus Start

This topic shows detailed information about “Record Trace - IFCID 174 - Arch Log CMD Sus Start”.

Record trace - IFCID 174 - Arch Log CMD Sus Start

The field labels shown in the following sample layout of “Record Trace - IFCID 174 - Arch Log CMD Sus Start” are described in the following section.

ACE: 1

QW0174EB X'024391B8'  QW0174UR X'0242C168'

Field Name: QW0174AC

ACE

The relative number of the agent control element address in the ACE cross-reference table. This table is printed at the end of each location for every trace specified.

Field Name: QW0174AC
IFCID 175 - Arch Log CMD Sus End

This topic shows detailed information about “Record Trace - IFCID 175 - Arch Log CMD Sus End”.

Record trace - IFCID 175 - Arch Log CMD Sus End

The field labels shown in the following sample layout of “Record Trace - IFCID 175 - Arch Log CMD Sus End” are described in the following section.

ACE: 1
QW0175EB X'024391B8' QW0175UR X'0242C168'

ACE

The relative number of the agent control element address in the ACE cross-reference table. This table is printed at the end of each location for every trace specified.

Field Name: QW0175AC
IFCID 177 - Package Allocation

This topic shows detailed information about "Record Trace - IFCID 177 - Package Allocation".

Record trace - IFCID 177 - Package Allocation

The field labels shown in the following sample layout of "Record Trace - IFCID 177 - Package Allocation" are described in the following section.

Package NetworkID: G91E81D0 LUNAME: D179 LUWSEQ: 2
Allocation Requesting Location: 9.30.129.208
Requesting Timestamp: N/P
AR Name: gixxer PRDID: CLNT/1ER V8 R1 M4
ACCTKX: X'C7F9F1C5F8F1C4F04BC4F1F7F9000F92022652404040'
LOCATION: DSND81B
COLLECTION ID: AIXPGMS
PACKAGE ID: SQLEMT
CONSISTENCY TOKEN: X'5A4276344E644L54'
VERSION NAME: N/P
DYNAMICRULES: RUN
PLAN: DISTSERV
ISOLATION: CS
ACQUIRE: USE
RELEASE: COMMIT
REOPTIMIZATION: NO
DEFERPrepare: NO
KEEPDYNAMIC: NO
BBPROTOCOL: DRDA
OPT_HINT_IDENT: 'BLANK'
IMMEDWRITE: NO

LOCATION

The location of the package. This field shows 'BLANK' if the local location is not defined.
Field Name: QW0177LO

COLLECTION ID

The collection name.
Field Name: QW0177CO

PACKAGE ID

The package identifier.
Field Name: QW0177PI

CONSISTENCY TOKEN

The consistency token (timestamp) of the program.
Field Name: QW0177CT

VERSION NAME

The version. This field shows N/P if the record does not contain a valid version.
Field Name: QW0177VN

DYNAMICRULES

The value of the DYNAMICRULES option on the BIND/REBIND command:
RUN  Runtime rules apply to a dynamic SQL statement for authorization checking and object qualification at run time.

BIND  Bind-time rules apply to a dynamic SQL statement for authorization checking and object qualification at run time.

N/P in this field indicates that DYNAMICRULES was not specified.

Field Name: QW0177DY

PLAN

The name of the plan under which the package is running.

Field Name: QW0177PL

ISOLATION

The isolation level of the package:

RR  Repeatable read
CS  Cursor stability
RS  Read stability
UR  Uncommitted read

Field Name: QW0177IS

ACQUIRE

The acquire level of the package.

Field Name: QW0177AQ

RELEASE

The release level of the package.

Field Name: QW0177RL

REOPTIMIZATION

Indicates whether reoptimization was requested:

YES  REOPT(VARS) was specified to reoptimize the access path of the SQL statement at run time.

NO  NOREOPT(VARS) was specified to optimize the access path of the SQL statement only at bind time.

Field Name: QW0177RO

DEFERPREPARE

Indicates whether the preparation of dynamic SQL statements was deferred:

YES  DEFER(PREPARE) was specified to defer the preparation of the dynamic SQL statements that refer to remote objects until run time.

NO  NODEFER(PREPARE) was specified to prepare the dynamic SQL statements at bind time.

Field Name: QW0177DP

KEEPDYNAMIC

Indicates whether DB2 keeps (KEEPDYNAMIC(YES)) or discards (KEEPDYNAMIC(NO)) prepared SQL statements at commit points.
FIELD NAME: QW0177KD

DBPROTOCOL
Protocol. Possible values are:

DRDA
PRIVATE

FIELD NAME: QW0177PR

OPT_HINT_IDENT
Query optimization hint identifier, the default is blanks.

FIELD NAME: QW0177OH

IMMEDWRITE
Indicates how DB2 updates group buffer pool dependent pages. This is only valid in a data-sharing environment.
Group buffer pool dependent pages can be written out to DASD or SYSTEM pagesets. Values shown are:

NO  DB2 uses normal write activity for updates, this is the default.
     Pages are written out at, or before phase 2 commit, or at the end of an abort for transactions that have rolled back.

PH1  Pages are written out at, or before phase 1 commit.
     If a transaction subsequently rolls back, the pages are updated in the group buffer pool at the end of the rollback, and are written out at the end of the abort.

YES  Pages are written out to the coupling facility as soon as the buffer update commits. Pages are written out regardless of whether the update occurs during forward progress or rollback of the transaction.
     This option can affect DB2 performance due to coupling facility overhead.

N/P  The DB2 subsystem is not part of a data sharing group.

FIELD NAME: QW0177WI
This topic shows detailed information about “Record Trace - IFCID 178 - IBM Service Record”.

This record is for IBM service use.
This topic shows detailed information about “Record Trace - IFCID 179 - IBM Service Record”.

This record is for IBM service use.
IFCID 180 - DC Communication Buffers

This topic shows detailed information about “Record Trace - IFCID 180 - DC Communication Buffers”.

The format of this record depends on the format of the network protocol.

Record trace - IFCID 180 - DC Communication Buffers

The field labels shown in the following sample layout of “Record Trace - IFCID 180 - DC Communication Buffers” are described in the following section.

**If SNA:**

-------

TYPE OF EVENT: SENT AN FMH-5 TO ALLOCATE A CONVERSATION
NETWORK PROTOCOL: SNA
CONVERSATION ID: X'00000010'
SESSION: X'1000000000000000'
MSG LENGTH: 52
MSG/FMH-5: TRACEXXXX1XXXXXXXXX2XXXXXXXXX3XXXXXXXXX4XXXXXXXXXZ

**If TCP/IP V4:**

-------------

TYPE OF EVENT: RECEIVED DISTRIBUTED DATA MESSAGE
NETWORK PROTOCOL: TCP/IP V4
SOCKET DESCRIPTR: X'000000010'
LOCAL PORT: X'A7A7'
PARTNER PORT: X'A7A7'
IP ADDRESS: X'1000000000000000D3D4E7E7E7E7E7E9'
MSG LENGTH: 52
MSG/FMH-5: TRACEXXXX1XXXXXXXXX2XXXXXXXXX3XXXXXXXXX4XXXXXXXXXZ

**If TCP/IP V6:**

-------------

TYPE OF EVENT: RECEIVED A DISTRIBUTED DATA MESSAGE
NETWORK PROTOCOL: TCP/IP V6
SOCKET DESCRIPTR: X'000000010'
LOCAL PORT: X'A7A7'
PARTNER PORT: X'A7A7'
IP ADDRESS: X'1000000000000000D3D4E7E7E7E7E7E9'
MSG LENGTH: 52
MSG/FMH-5: TRACEXXXX1XXXXXXXXX2XXXXXXXXX3XXXXXXXXX4XXXXXXXXXZ

**TYPE OF EVENT**

The type of event:

- **R** A distributed data message was received.
- **S** A distributed data message was sent.
- **F** An FMH-5 on an incoming conversation was received (only possible for VTAM conversations).
- **A** An FMH-5 to allocate a conversation was sent (only possible for VTAM conversations).

**Field Name:** QW0180E

**NETWORK PROTOCOL**

The type of network protocol:

- SNA
- TCP/IP IPV4
- TCP/IP IPV6

**Field Name:** QW0180NP

**SESSION ID or IP ADDRESS(V4), LOCAL PORT(V4), PARTNER PORT(V4)**

For SNA: this field contains the session ID. For TCP/IP: If QWHSRN is lower than 91, this field contains the 32-bit IPV4 IP address, followed by the 16-bit local port number, followed by the 16-bit partner port number.

**Field Name:** QW0180SI
IFCID 180 - DC Communication Buffers

IPADDRESS (V6)

The IP address for TCP/IP:
- If QWHSRN is lower than or equal to 91, this field contains the IP address in internal form.
- If QW0180NP is equal to ‘01’B, this field contains an IPV4 IP address, which can be mapped.
- If QW0180NP is equal to ‘10’B, this field contains a 128-bit IPV6 IP address.

Field Name: QW0180IP

MODE

For SNA: the entry name of the log mode.

Field Name: QW0180LM

LOCAL PORT (V6)

The local port.

Field Name: QW0180LP

PARTNER PORT (V6)

The partner port.

Field Name: QW0180PP

MSG LENGTH

The length of the variable length area mapped by QW0180DS.

Field Name: QW0180DL

MSG/FMH-5

The variable length message or FMH-5 data. (The password in the FMH-5 or the TCP/IP message is changed to blanks.)

Field Name: QW0180DS
IHFID 181 - IBM Service Record

This topic shows detailed information about “Record Trace - IFCID 181 - IBM Service Record”.

This record is for IBM service use.
This topic shows detailed information about “Record Trace - IFCID 182 - IBM Service Record”.

This record is for IBM service use.
IFCID 183 - DRDS RDS/SCC Interface

This topic shows detailed information about “Record Trace - IFCID 183 - DRDS RDS/SCC Interface”.

This record provides information about the type of request being processed at the requester.

This record is produced only for DRDA requests.

Record trace - IFCID 183 - DRDS RDS/SCC Interface

The field labels shown in the following sample layout of “Record Trace - IFCID 183 - DRDS RDS/SCC Interface” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRDS RDS/SCC 'BLANK'</td>
<td>INTERFACE NETWORKID: DEIBMIPS LUNAME: IPSAR721 LUWSEQ: 1</td>
</tr>
<tr>
<td>EVENT TYPE</td>
<td>RETURN TO RDS/SCC</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>CONNECT RESET FOR BIND</td>
</tr>
<tr>
<td>LOCATION</td>
<td>LOCATIONNAME0002</td>
</tr>
<tr>
<td>COLLECTION</td>
<td>COLLECTIONNAME0002</td>
</tr>
<tr>
<td>PACKAGE ID</td>
<td>PROGNAM2</td>
</tr>
<tr>
<td>SQL STATEMENT NUMBER</td>
<td>0</td>
</tr>
<tr>
<td>SQL STATEMENT TYPE</td>
<td>80 RETCODE: 77</td>
</tr>
</tbody>
</table>

EVENT TYPE
The type of event.
Field Name: QW0183E

FUNCTION
The type of function.
Field Name: QW0183FN

LOCATION
The location name of the application server.
Field Name: QW0183LN

COLLECTION
The collection name.
Field Name: QW0183CO

PACKAGE ID
The package ID.
Field Name: QW0183PN

SQL STATEMENT TYPE
The SQL statement type:

- 003 OPEN
- 004 FETCH
- 005 CLOSE
- 014 PREPARE
- 015 EXECUTE
- 016 EXECUTE IMMEDIATE
IFCID 183 - DRDS RDS/SCC Interface

Field Name: QW0183ST

SQL STATEMENT NUMBER

The SQL statement number.

Field Name: QW0183SN

RETURN CODE

The return code.

Field Name: QW0183RC
IFCID 184 - DC Communication Buffers

This topic shows detailed information about “Record Trace - IFCID 184 - DC Communication Buffers”.

Record trace - IFCID 184 - DC Communication Buffers

The field labels shown in the following sample layout of “Record Trace - IFCID 184 - DC Communication Buffers” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>TYPE OF EVENT</td>
<td>RECEIVED A DISTRIBUTED DATA MESSAGE</td>
</tr>
<tr>
<td>NETWORK PROTOCOL</td>
<td>TCP/IP V4</td>
</tr>
<tr>
<td>LOCAL PORT</td>
<td>X'C9D5'</td>
</tr>
<tr>
<td>PARTNER PORT</td>
<td>X'C6D6'</td>
</tr>
<tr>
<td>MSG LENGTH</td>
<td>255</td>
</tr>
<tr>
<td>MESSAGE TEXT</td>
<td></td>
</tr>
</tbody>
</table>

**TYPE OF EVENT**

The type of event can be one of the following:

- MSG RECEIVED
- MSG SENT
- MSG CONTINUED

Field Name: QW0184E

**NETWORK PROTOCOL**

The partner port.

Field Name: QW0184PP

**SOCKET DESCRIPTOR**

The descriptor of the TCP/IP socket in hexadecimal.

Field Name: QW0184SD

**IP ADDRESS**

The IP Address or Port in hexadecimal.

Field Name: QW0184SI

**LOCAL PORT**

The local port.

Field Name: QW0184LP

**PARTNER PORT**

The partner port.

Field Name: QW0184PP

**MESSAGE LENGTH**

The length of the message.

Field Name: QW0184DL

**MESSAGE TEXT**

The text of the message.

Field Name: QW0184DS
This topic shows detailed information about “Record Trace - IFCID 185 - READs Data Capture Start”.

When present, data is printed in hexadecimal dump format, otherwise NO DATA is printed.
IFCID 186 - IBM Service Record

This topic shows detailed information about “Record Trace - IFCID 186 - IBM Service Record”.

This record is for IBM service use.
IFCID 188 - READs Data Capture End

Record trace - IFCID 188 - READs Data Capture End

The field labels shown in the following sample layout of “Record Trace - IFCID 188 - READs Data Capture End” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>REQUEST TYPE</td>
<td>The type of request from the WQALCDCD field of the IFI qualification area.</td>
</tr>
<tr>
<td>READS REQUEST FLAG</td>
<td>The reads request flag. If the value is x'40', reads were required because more data was available than would fit in the user return area. If this occurs frequently, consider increasing the size of the user return area.</td>
</tr>
<tr>
<td>DESCRIBES</td>
<td>The number of data capture describes.</td>
</tr>
<tr>
<td>LONGEST LOG READ</td>
<td>The portion of the log read that took the longest amount of time.</td>
</tr>
<tr>
<td>BEGIN REQUEST RBA</td>
<td>The beginning RBA of the requested log range.</td>
</tr>
<tr>
<td>DESCRIBE ELAPSED</td>
<td>The elapsed time of the data capture describe.</td>
</tr>
<tr>
<td>LOG READS RETURNED</td>
<td>The total number of log records from which data rows are returned for this single READs request.</td>
</tr>
<tr>
<td>END REQUEST RBA</td>
<td>The end RBA of the requested log range.</td>
</tr>
</tbody>
</table>
The number of log reads performed.
Field Name: QW0188LR

TABLES RETURNED
The number of data capture tables returned.
Field Name: QW0188TB

LOG READ ELAPSED
The elapsed time of the longest log read.
Field Name: QW0188LL

LOG RECS RETURNED
The total number of log records retrieved by one or more reads requests for IFCID 185 for a single SQL change. If the value in this field is less than the value in LOG RECS CAPTURED, then additional log records must be retrieved to obtain all log records involved in the SQL change.
Field Name: QW0188RR

DATA ROWS RETURNED
The number of data rows returned.
Field Name: QW0188DR

LOG EXTRACT ELAPSED
The log extraction elapsed time for IFCID 185 requests.
Field Name: QW0188LT

LOG RECS CAPTURED
The total number of log records captured on the log for this particular SQL change.
Field Name: QW0188LC

DATA DESCR. RET
The number of data descriptions returned.
Field Name: QW0188DD
IFCID 190 - IBM Service Record

This topic shows detailed information about “Record Trace - IFCID 190 - IBM Service Record”.

This record is for IBM service use.
This record can contain six types of data section. One header section and one 6B DSS section are always present. The other sections are only printed if they are present.

“IFCID 191 - Command and/or Reply Section” on page 40-582
This topic shows detailed information about “Record Trace - IFCID 191 - Command and/or Reply Section”.

“IFCID 191 - DB2 ZEDA” on page 40-584
This topic shows detailed information about “Record Trace - IFCID 191 - DB2 ZEDA”.

“IFCID 191 - Header Section” on page 40-585
This topic shows detailed information about “Record Trace - IFCID 191 - Header Section”.

“IFCID 191 - Late Descriptor Section” on page 40-587
This topic shows detailed information about “Record Trace - IFCID 191 - Late Descriptor Section”.
IFCID 191 - Command and/or Reply Section

IFCID 191 - Command and/or Reply Section

This topic shows detailed information about “Record Trace - IFCID 191 - Command and/or Reply Section”.

Record trace - IFCID 191 - Command and/or Reply Section

The field labels shown in the following sample layout of “Record Trace - IFCID 191 - Command and/or Reply Section” are described in the following section.

EYECATCHER

The type of data in this section:

- DRDACMND: Command data
- DRDARPLY: Reply data
- DRDAHCMND: Command data
- DRDAHRPY: Reply data
- DRDARDTA: Reply data
- DRDACDTA: Command data

Field Name: QW0191RE

PARSE STATUS

The parse status:

- DRDASUCC: The parse is successful.
- DRDAFAIL: The parse is unsuccessful.

Field Name: QW0191PS

CODE POINT

The code point.

Field Name: QW0191C3

RELATIVE NUMBER

The relative number of the data stream structure carrier.

Field Name: QW0191NM

RPY/RQS/OBJ

Offset to the start of RPY/RQS/OBJ DSS within the IFCID 191 record.

Field Name: QW0191OF

PARSE FAIL
IFCID 191 - Command and/or Reply Section

Offset relative to the IFCID 191 record point at which parse failed.
Field Name: QW0191FO

FD SECTION
Offset relative to the IFCID 191 record to the LATE DESCRIPTOR section.
Field Name: QW0191D1

RT SECTION
Offset relative to the IFCID 191 record to the RDTA DATA section.
Field Name: QW0191D2

LT SECTION
Offset relative to the IFCID 191 record to the FD LIDLST section.
Field Name: QW0191D3

EA SECTION
Offset relative to the IFCID 191 record to the ZEDA DATA section.
Field Name: QW0191D4
**Record trace - IFCID 191 - DB2 ZEDA**

The field labels shown in the following sample layout of “Record Trace - IFCID 191 - DB2 ZEDA” are described in the following section.

<table>
<thead>
<tr>
<th>EYECATCHER</th>
<th>DRDAZEDA</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000 000008E8</td>
<td>E9C5E1 00010000 00000000 00000488 04880488 04880000 00000000 00000000 00000000</td>
</tr>
<tr>
<td>0020 00000220 202AE0B8</td>
<td>00000000 00000000 00000000 00000000 00000000 00000000</td>
</tr>
<tr>
<td>0040 00000000</td>
<td>00000000 00000000 00000000 00000000 00000000 00000000</td>
</tr>
<tr>
<td>0060 00000000</td>
<td>00000000 00000000 00000000 00000000 E3C800 00503391 00000000 00000000</td>
</tr>
<tr>
<td>0080 00000000</td>
<td>00000000 00000000 00000000 00000000 00000000 00000000 00000000</td>
</tr>
<tr>
<td>00A0 00000000</td>
<td>00000000 00000000 00000000 00000000 10000040 00000000 00000000 00000000</td>
</tr>
<tr>
<td>00C0 00000000</td>
<td>00000025</td>
</tr>
<tr>
<td>00E0 00000000</td>
<td>00000000 00000000 00000000 00000000 00000000 00000000 00000000</td>
</tr>
</tbody>
</table>

**EYECATCHER**

The type of data in this section:

- **DRDACMND**
  Command data

- **DRDARPLY**
  Reply data

- **DRDAHCMND**
  Command data

- **DRDAHRPY**
  Reply data

- **DRDARDTA**
  Reply data

- **DRDACDTA**
  Command data

**Field Name:** QW0191RE
IFCID 191 - Header Section

This topic shows detailed information about “Record Trace - IFCID 191 - Header Section”.

Record trace - IFCID 191 - Header Section

The field labels shown in the following sample layout of “Record Trace - IFCID 191 - Header Section” are described in the following section.

LOCATION

The name of the remote location.

Field Name: QW0191LN

VERSION

The version number for all sections.

Field Name: QW0191VS

OBJ LEN.

The length of the failed object.

Field Name: QW0191FL

REASON

The reason code.

Field Name: QW0191RS

RECORD

The sequence number for this IFCID 191 record out of the total number of IFCID 191 records.

Field Name: QW0191NO

OF TOTAL

The total number of IFCID 191 records.

Field Name: QW0191TO

MODULE

The module name.

Field Name: QW0191MN

SOURCE

The source ID in the module.

Field Name: QW0191MI

ERROR TOKEN

The unique error token.

Field Name: QW0191TK

DDM COMMAND CODE POINT

Chapter 40. IFCID Record Blocks  40-585
IFCID 191 - Header Section

The DDM command code point.

Field Name: QW0191C1

DB2 PARSE STATE

The DB2 parse state:

P1 Application requester parse
P2 Application server parse

Field Name: QW0191PA

RN RECEIVED

The number of relay messages received.

Field Name: QW0191RN

OBJDSS RECEIVED

The number of object data stream structures received.

Field Name: QW0191ON

DSS TOTAL

The total number of data stream structures.

Field Name: QW0191DN

ERROR TYPE

The type of error:

0 SQLSTATE is SQLCA generated
1 Reply message sent

Field Name: QW0191ER

DIMENSION

The dimension of PARSE TRACE ARRAY.

Field Name: QW0191TN

PARSE TRACE ARRAY

The last five top level parse traces. These are shown in the format STATE, EVENTS.

Field Name: QW0191PT
IFCID 191 - Late Descriptor Section

This topic shows detailed information about “Record Trace - IFCID 191 - Late Descriptor Section”.

Record trace - IFCID 191 - Late Descriptor Section

The field labels shown in the following sample layout of “Record Trace - IFCID 191 - Late Descriptor Section” are described in the following section.

LATE DESCRIPTORS PROCESSED
The number of late environmental descriptors processed.
Field Name: QW0191LD

SQLDTAGRP TRIPLETS
The total number of data stream structures.
Field Name: QW0191GN

L1
SQLDTAGRP local ID extracted.
Field Name: QW0191L1

L2
SQLCADTA local ID extracted.
Field Name: QW0191L2

L3
SQLDTA local ID extracted.
Field Name: QW0191L3

L4
SQLDTARD local ID extracted.
Field Name: QW0191L4

GEOMETRY STATUS
The FD:OCA geometry status. This field is a bit mask. The hexadecimal value of the field is printed.
- If bit 0 is on, the status of SQLDTAGRP is OK.
- If bit 1 is on, the status of SQLCADTA is OK.
- If bit 2 is on, the status of SQLDTA is OK.
- If bit 3 is on, the status of SQLDTARD is OK.
Field Name: QW0191GO
IFCID 191 - 6B DSS Section

This topic shows detailed information about “Record Trace - IFCID 191 - 6B DSS Section”.

Record trace - IFCID 191 - 6B DSS Section

The field labels shown in the following sample layout of “Record Trace - IFCID 191 - 6B DSS Section” are described in the following section.

| Field Name: QW0191RE |
IFCID 192 - DDM Level 6A Header Errors

DDM level 6A header errors show the data from IFCID 192.

"IFCID 192 - Current 6A Header" on page 40-590
This topic shows detailed information about "Record Trace - IFCID 192 - Current 6A Header".

"IFCID 192 - DDM Level 6A Header Errors" on page 40-591
This topic shows detailed information about "Record Trace - IFCID 192 - DDM Level 6A Header Errors".

"IFCID 192 - Previous 6A Header" on page 40-592
This topic shows detailed information about "Record Trace - IFCID 192 - Previous 6A Header".
IFCID 192 - Current 6A Header

This topic shows detailed information about “Record Trace - IFCID 192 - Current 6A Header”.

OFFSET
Offset into the data stream of the current DDM level 6A header (that is, the invalid DDM header).
Field Name: QW0192CO

GDS LENGTH
Generalized data stream (GDS) length field.
Field Name: QW0192CL

DDM CONST
The DDM constant.
Field Name: QW0192CI

FLAG
The DDM flag byte.
Field Name: QW0192CF

REQ CORR
The DDM request correlator.
Field Name: QW0192CC
IFCID 192 - DDM Level 6A Header Errors

This topic shows detailed information about “Record Trace - IFCID 192 - DDM Level 6A Header Errors”.

Record trace - IFCID 192 - DDM Level 6A Header Errors

The field labels shown in the following sample layout of “Record Trace - IFCID 192 - DDM Level 6A Header Errors” are described in the following section.

REMOTE LOCATION
The name of the remote location.
Field Name: QW0192LN

VERSION NUMBER
The version number for the IFCID 192 records.
Field Name: QW0192VN

CSECT
The CSECT that detected the error.
Field Name: QW0192CS

ERROR TYPE
The DDM error type returned.
Field Name: QW0192ER

SEVERITY
The DDM severity code returned.
Field Name: QW0192SV

ERROR CODE
The DDM error code returned. For DDM protocol errors, this is the DDM PRCCNVCD value. For DDM syntax errors, this is the DDM SYNERRCD value.
Field Name: QW0192CD
IFCID 192 - Previous 6A Header

IFCID 192 - Previous 6A Header
This topic shows detailed information about “Record Trace - IFCID 192 - Previous 6A Header”.

OFFSET
Offset into the data stream of the current DDM level 6A header (that is, the last valid DDM header).
Field Name: QW0192PO

GDS LENGTH
Generalized data stream (GDS) length field.
Field Name: QW0192PL

DDM CONST
The DDM constant.
Field Name: QW0192PI

FLAG
The DDM flag byte.
Field Name: QW0192PF

REQ CORR
The DDM request correlator.
Field Name: QW0192PC
IFCID 193 - UOW/SQLCODE Mismatch

This topic shows detailed information about “Record Trace - IFCID 193 - UOW/SQLCODE Mismatch”.

Record trace - IFCID 193 - UOW/SQLCODE Mismatch

The field labels shown in the following sample layout of “Record Trace - IFCID 193 - UOW/SQLCODE Mismatch” are described in the following section.

REMOTE LOCATION

The location name of the server.

Field Name: QW0193LN

VERSION

The version number of this trace record.

Field Name: QW0193VS

CSECT

The CSECT that detected the error.

Field Name: QW0193CS

SQLCODE

The SQL code returned by the server.

Field Name: QW0193SC

COMMAND SENT

The command sent to the server.

Field Name: QW0193CO

UOW DISPOSITION

The unit of work (UOW) disposition reported by the server.

Field Name: QW0193UW
IFCID 194 - Invalid SNA FMH-5 Received

This topic shows detailed information about “Record Trace - IFCID 194 - Invalid SNA FMH-5 Received”.

Record trace - IFCID 194 - Invalid SNA FMH-5 Received

The field labels shown in the following sample layout of “Record Trace - IFCID 194 - Invalid SNA FMH-5 Received” are described in the following section.

REMOTE LOCATION: SYD2
VERSION NUMBER: 1
CSECT: TDG
SNA SENSE CODE: X'E3C4C740'
FMH5 DATA:
0000 E3C5E2E3 40F14040

REMOTE LOCATION
The name of the remote location.
Field Name: QW0194LN

VERSION NUMBER
The version number of this trace record.
Field Name: QW0194VN

CSECT
The CSECT that detected the error.
Field Name: QW0194CS

SNA SENSE CODE
The SNA sense code describing the error.
Field Name: QW0194SN

FMH5 DATA
The invalid SNA FMH-5 record.
Field Name: QW0194DS
IFCID 195 - SQLDA Discrepancy

This topic shows detailed information about “Record Trace - IFCID 195 - SQLDA Discrepancy”.

Record trace - IFCID 195 - SQLDA Discrepancy

The field labels shown in the following sample layout of “Record Trace - IFCID 195 - SQLDA Discrepancy” are described in the following section.

REMOTE LOCATION
The name of the remote location.
Field Name: QW0195LN

VERSION
The version number of this trace record.
Field Name: QW0195VI

MODULE
The name of the module.
Field Name: QW0195MN

ID
The source ID in the module.
Field Name: QW0195UI

FIELD IN ERROR
The field in error:
SQLD  The number of entries in SQLD
SQLTYPE  Data type
SQLLEN  Data length
SQLDATA  CCSID
Field Name: QW0195FD

COLUMN
The column number for the field in error.
Field Name: QW0195NO

EXISTING SQLDA
The contents in the existing SQLDA field.
Field Name: QW0195SE

NEW SQLDA
The contents in the new SQLDA field.
IFCID 195 - SQLDA Discrepancy

Field Name: QW0195SN
IFCID 196 - Timeout Data

IFCID 196 provides information on a lock request that resulted in the timeout of its DB2 task because one or more other tasks were holding incompatible locks on the requested resource. DB2 always obtains (GETMAINS) storage for this record even if the user did not activate Statistics trace class 3 or performance trace class 6.

"IFCID 196 - Holder" on page 40-598
This topic shows detailed information about “Record Trace - IFCID 196 - Holder”.

"IFCID 196 - Timeout Header” on page 40-601
This topic shows detailed information about “Record Trace - IFCID 196 - Timeout Header”.

IFCID 196 - Holder

IFCID 196 - Holder

This topic shows detailed information about “Record Trace - IFCID 196 - Holder”.

Record trace - IFCID 196 - Holder

The field labels shown in the following sample layout of “Record Trace - IFCID 196 - Holder” are described in the following section.

**PRIMAUTH**

The authorization ID of the thread holding the resource.

**Field Name:** QW0196HB

**PLAN NAME**

The holder’s plan name or, if there is contention with a retained lock, the word SYSTEM.

**Field Name:** QW0196HP

**CORR ID**

The holder’s correlation ID or, if there is contention with a retained lock, the word SYSTEM.

**Field Name:** QW0196HR

**CONN**

The holder’s connection ID or, if there is contention with a retained lock, the word SYSTEM.

**Field Name:** QW0196HN

**LUWID - NETWORKID, LUNAME, INSTANCE**

This field contains an asterisk (*) if the lock holder is not a database access thread (DBAT). It provides the input for the:

- Holder’s network ID or, if there is contention with a retained lock, the word SYSTEM.
- Holder’s LU name or, if there is contention with a retained lock, the word SYSTEM.
- Holder’s LUW instance or, if there is contention with a retained lock, the word SYSTEM.

**Note:** This field is only valid for distributed threads.

**Field Name:** QW0196HL

**OWNING WORK UNIT**

The holder’s owning work unit. This value is printed in hexadecimal.

If there is contention with a retained lock, this field is set to X'00'.

**Field Name:** QW0196HO

**LOCK STATE**
**IFCID 196 - Holder**

The holder's lock state.

**Field Name:** QW0196HS

**LOCK DURATION**

The lock duration of the holder:

- **MANUAL**  Varies depending on the ISOLATION parameter (QW0196HD=x'20')
- **MANUAL+1**  Temporary change of consistency level from CS to RR during bind and DDL (QW0196HD=x'21')
- **COMMIT**  Until commit (QW0196HD=x'40')
- **COMMIT+1**  Past commit; applies to locks needed to maintain the position for a cursor opened WITH HOLD (QW0196HD=x'41')

**ALLOCATION**

Until deallocation (QW0196HD=x'60')

- **PLAN**  For the duration of the plan (QW0196HD=x'80')
- **UTIL**  For the duration of the utility execution (QW0196HD=x'81')
- **INTEREST**  Duration used for P-locks (QW0196HD=x'FE')

- **FREE ALL**  Until all locks are freed (QW0196HD=x'FF')
- **x'00'**  Contention with a retained lock (QW0196HD=x'00')

**Field Name:** QW0196HD

**MEMBER**

The holder's DB2 member name. For non-data sharing environments, N/P is shown in this field.

**Field Name:** QW0196HI

**TRANSACT**

The transaction or application name that is run.

**Field Name:** QWHCEUTX

**WS_NAME**

The end user's workstation name.

**Field Name:** QWHCEUWN

**END_USER**

The user ID of the workstation end user. This user ID can be different from the authorization ID used to connect to DB2. This field contains blanks if the client does not supply this information.

**Field Name:** QWHCEUID

**STMT ID**

The cached statement ID for the statement holding the resource. A value of zero indicates that the client did not supply this information.
IFCID 196 - Holder

Field Name: QW0196H9

STMT TYPE

STATIC
   The statement is of type static.

DYNAMIC
   The statement is of type dynamic.

Field Name: QW0196HY
IFCID 196 - Timeout Header

This topic shows detailed information about “Record Trace - IFCID 196 - Timeout Header”.

Record trace - IFCID 196 - Timeout Header

The field labels shown in the following sample layout of “Record Trace - IFCID 196 - Timeout Header” are described in the following section.

NUMBER OF HOLDERS/WAITERS

The number of agents causing the timeout.

Field Name: QW0196NU

LOCK HASH VALUE

The hash value of the locked resource.

Field Name: QW0196RH

LOCK RES TYPE

The locked resource type.

Note: For data sharing, SKELETON CURSOR TABLE LOCKING and SKELETON PACKAGE TABLE LOCK are LP-locks (an LP-lock has an L-lock component and a P-lock component).

Field Name: QW0196RN

DBID

The database ID. This field is not applicable if the value in LOCK RES TYPE is SKELETON CURSOR TABLE LOCKING, SKELETON PACKAGE TABLE LOCK, COLLECTION, or ALTER BUFFER POOL.

Deduced from the DB2 fields QW0196KD, QW0105DN or QW0107DN.

If present, the database name is shown, otherwise the decimal identifier from QW0196KD is shown or N/A if this value is 0.

Field Name: RT0196DB

OBID

The object ID. This field is not applicable if the value in LOCK RES TYPE is SKELETON CURSOR TABLE LOCKING, SKELETON PACKAGE TABLE LOCK, COLLECTION, or ALTER BUFFER POOL.

Deduced from the DB2 fields QW0196KP, QW0105TN or QW0107TN.

If present, the object name is shown, otherwise the decimal identifier from QW0196KP is shown or N/A if this value is 0.

Field Name: RT0196OB

REQUESTED FUNCTION

The victim's type of request.

Field Name: QW0196WU
IFCID 196 - Timeout Header

REQUESTED STATE

The victim's lock state.

Field Name: QW0196WS

REQUESTED DURATION

The victim's lock duration.

Field Name: QW0196WD

REQUESTED FLAGS

The victim's lock flag.

Field Name: QW0196WF

REQUESTED OWNING WORK UNIT

The victim's owning work unit.

Field Name: QW0196WO

ZPARAM INTERVAL

The timeout interval of the ZPARAM value.

Field Name: QW0196TI

INTERVAL COUNTER

The timeout counter for this thread.

Field Name: QW0196TC

WTR STMT ID

The cached statement ID for the statement waiting for the resource. A value of zero indicates that the client did not supply this information.

Field Name: QW0196W9

WTR STMT TYPE

The waiter statement information. Possible values are:

STATIC

The statement is of type static.

DYNAMIC

The statement is of type dynamic.

Field Name: QW0196WY
This topic shows detailed information about “Record Trace - IFCID 197 - DB2 Messages”.

This IFCID enables monitoring of DB2 messages. If this trace is enabled, all console messages will be written to IFCID 197 records. This record is written when performance trace class 18 is on.

Note: Not all messages are written to the message log. DISPLAY commands, for example, are filtered out and are not shown in the log.

Record trace - IFCID 197 - DB2 Messages

The field labels shown in the following sample layout of “Record Trace - IFCID 197 - DB2 Messages” are described in the following section.

MESSAGE ID

The message ID.

Field Name: QW0197ID

MESSAGE Text

The message text can consist of up to 2500 bytes.

Field Name: QW0197TX
IFCID 198 - Buffer Manager Page Access

Record Trace - IFCID 198 - Buffer Manager Page Access

The field labels shown in the following sample layout of “Record Trace - IFCID 198 - Buffer Manager Page Access” are described in the following section.

DBID: 6  PSID: 112  BPID: X'00'

**FUNCTION**

GET PAGE

**PAGE STATUS**

PAGE HIT IN BUFFERPOOL

**ACCESS**

RANDOM

**PAGE REFRESH**

N/A

**PARTITION**

0

**DBID**

The database ID. Deduced from the DB2 fields QW0198DB, QW0105DN or QW0107DN.

When present, the database name is shown, otherwise the decimal identifier from QW0198DB is shown or N/A if this value is 0.

**Field Name:** RT0198OB

**PSID**

The page set object identifier. When present, this is the page set object name, otherwise the decimal identifier from QW0198OB is shown.

**Field Name:** RT0198PS

**BPID**

The buffer pool identifier.

**Field Name:** QW0198BP

**FUNCTION**

The page request function code.

**Field Name:** QW0198FC

**PAGE STATUS**

The page status in the buffer pool.

**Field Name:** QW0198PS

**ACCESS**

The page access type. This is only applicable if the value in FUNCTION is GET PAGE or RELEASE PAGE.

**Field Name:** QW0198AT

**PAGE**

The page number.

**Field Name:** QW0198PN

**ACE**

The relative number of the agent control element address in the ACE cross-reference table. This table is printed at the end of each location for every trace specified.
Field Name: QW0198AC

PAGE REFRESH
Page refresh status in case of a missed page in the virtual buffer pool.
Possible values:
• FROM HIPERPOOL
• FROM GROUP BUFFER POOL
• FROM DASD

Field Name: QW0198PR

PARTITION
The partition number. This field contains 0 if the request is non-partitioned.

Field Name: QW0198PT
IFCID 199 - Buffer Pool Statistics at Data Set Level

This topic shows detailed information about “Record Trace - IFCID 199 - Buffer Pool Statistics at Data Set Level”.

Record trace - IFCID 199 - Buffer Pool Statistics at Data Set Level

The field labels shown in the following sample layout of “Record Trace - IFCID 199 - Buffer Pool Statistics at Data Set Level” are described in the following section.

INTERVAL DSSTIME COMPLETED

Stores the clock value at the end of the statistics interval.

Field Name: QW0199TS

DBID

Internal identifier of the database where the tablespace or indexspace resides.

The ID can be used to match column DBID of table SYSIBM.SYSDATABASE in the DB2 catalog.

Be aware the value in the catalog may have changed since the time the DB2 trace record was written.

Field Name: QW0199DB

DBNAME

Internal identifier of the database where the tablespace or indexspace resides.

The ID can be used to match column DBID of table SYSIBM.SYSDATABASE in the DB2 catalog.

Be aware the value in the catalog may have changed since the time the DB2 trace record was written.

Field Name: QW0199DB

GBP DEPENDENT

Indicates whether the pageset is group buffer pool dependent. This is possible only if DB2 has been set up for data sharing.

Field Name: QW0199GD

LAST IN SEQUENCE

Indicates if this is the last IFCID 0199 record.

Field Name: QW0199LS

OBID
IFCID 199 - Buffer Pool Statistics at Data Set Level

The internal identifier of the pageset. This can be either a table space or an index space.
For a table space the ID can be used to match column 'PSID' in SYSIBM.SYSTABLESPACE of the catalog.
For an index space the ID can be used to match column 'ISOBID' in SYSIBM.SYSINDEXES.
Be aware the value in the catalog may have changed since the time the DB2 trace record was written.

Field Name: QW0199OB

OBNAME

The internal identifier of the pageset. This can be either a table space or an index space.
For a table space the ID can be used to match column 'PSID' in SYSIBM.SYSTABLESPACE of the catalog.
For an index space the ID can be used to match column 'ISOBID' in SYSIBM.SYSINDEXES.
Be aware the value in the catalog may have changed since the time the DB2 trace record was written.

Field Name: QW0199OB

TYPE OF DATASET

Indicates whether the data set is a data table or an index space.

Field Name: QW0199ID

BPID

Identifies the buffer pool to which the information in this section refers:
• Values 0 through 49 are identifiers for BP0 through BP49.
• Values 80 through 89 are identifiers for BP32K through BP32K9.
• Values 100 through 109 are identifiers for BP8K through BP8K9.
• Values 120 through 129 are identifiers for BP16K through BP16K9.

Field Name: QW0199BP

SHADOW COPY

Indicates if it is a shadow data set:
1 Means this is a shadow data set.
0 Means this is not a shadow data set.

Field Name: QW0199SD

PARTITION

For a partitioned table space or index space, this is the partition number.
For a nonpartitioned table space or index space, this is the data set number.

Field Name: QW0199DN

AVG. DELAY I/O (MS) - SYNC.I/O FOR WRITE AND READ

Average synchronous I/O delay for pages in the pageset, in milliseconds.
IFCID 199 - Buffer Pool Statistics at Data Set Level

Field Name: QW0199SV
AVG. DELAY I/O (MS) - ASYNC.I/O FOR WRITE, READ, CASTOUT
Average asynchronous I/O delay for pages in the pageset, in milliseconds.
Field Name: QW0199AD
VPOOL CACHE CURR. - BUFFER POOL CACHED PAGES
Number of pageset pages in the virtual buffer pool.
Field Name: QW0199VP
MAX. DELAY I/O (MS) - SYNC.I/O FOR WRITE AND READ
Maximum synchronous I/O delay for pages in the pageset, in milliseconds.
Field Name: QW0199SX
MAX. DELAY I/O (MS) - ASYNC.I/O FOR WRITE, READ, CASTOUT
The maximum asynchronous I/O delay for pages in the pageset, in milliseconds.
Field Name: QW0199AX
VPOOL CACHED CHANGED - BUFFER POOL CACHED PAGES
Number of changed page set pages in the virtual buffer pool.
Field Name: QW0199VD
TOTAL I/O PAGES - SYNC.I/O FOR WRITE AND READ
The number of synchronous I/Os for the pageset in the reported interval.
Field Name: QW0199SP
TOTAL I/O PAGES - ASYNC.I/O FOR WRITE, READ, CASTOUT
Number of pages read or written asynchronously for the pageset in the reported interval.
Field Name: QW0199AP
TOTAL I/O COUNT
The number of asynchronous I/Os for the pageset during the reported interval.
Field Name: QW0199AC
CURRENT GETPAGES
The current number of Getpage requests.
Field Name: QW0199GP
This topic shows detailed information about “Record Trace - IFCID 201 - Alter Buffer Pool”.

This IFCID records the status of a buffer pool before and after an ALTER BUFFERPOOL command.

Record trace - IFCID 201 - Alter Buffer Pool

The field labels shown in the following sample layout of “Record Trace - IFCID 201 - Alter Buffer Pool” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BUFFERPOOL ID</td>
<td>Buffer pool internal identifier. The values 0 through 49 are the identifiers for BP0 through BP49. The values 80 through 89 are the identifiers for BP32K through BP32K9.</td>
</tr>
<tr>
<td>NAME</td>
<td>Buffer pool name.</td>
</tr>
<tr>
<td>ALTER COMMAND</td>
<td>This field indicates how the buffer pool was altered. Possible values: ALTER: The ALTER BUFFERPOOL command was used. AUTOSIZE: The AUTOMATIC AUTOSIZING (AUTOSIZE(YES) option on ALTER BUFFERPOOL was previously specified.</td>
</tr>
<tr>
<td>RETURN CODE</td>
<td>The return code for the ALTER BUFFERPOOL command.</td>
</tr>
<tr>
<td>REASON CODE</td>
<td>The reason code returned from an unsuccessful ALTER BUFFERPOOL command.</td>
</tr>
</tbody>
</table>

VPOOL SIZE (OLD):

<table>
<thead>
<tr>
<th>OLD STATUS</th>
<th>NEW STATUS</th>
</tr>
</thead>
<tbody>
<tr>
<td>VPOOL SIZE : 0</td>
<td>VPOOL SIZE : 10</td>
</tr>
<tr>
<td>VPOOL SEQ THRESH : 80</td>
<td>VPOOL SEQ THRESH : 80</td>
</tr>
<tr>
<td>VPOOL DWT THRESH : 30</td>
<td>VPOOL DWT THRESH : 30</td>
</tr>
<tr>
<td>VPOOL VDWT THRESH : 5</td>
<td>VPOOL VDWT THRESH : 5</td>
</tr>
<tr>
<td>PERCENTAGE : 5</td>
<td>PERCENTAGE : 5</td>
</tr>
<tr>
<td>BUFFERS : 0</td>
<td>BUFFERS : 0</td>
</tr>
<tr>
<td>VPOOL PLL SEQ THRESH : 50</td>
<td>VPOOL PLL SEQ THRESH : 50</td>
</tr>
<tr>
<td>ASSISTAN. SEQ THRESH : 0</td>
<td>ASSISTAN. SEQ THRESH : 0</td>
</tr>
<tr>
<td>PAGE STEAL METHOD : LRU</td>
<td>PAGE STEAL METHOD : LRU</td>
</tr>
<tr>
<td>AUTOSIZE : NO</td>
<td>AUTOSIZE : NO</td>
</tr>
<tr>
<td>FRAMESIZE : 4K</td>
<td>FRAMESIZE : 4K</td>
</tr>
<tr>
<td>VPOOL SIZE MIN : 0</td>
<td>VPOOL SIZE MIN : 0</td>
</tr>
<tr>
<td>VPOOL SIZE MAX : 0</td>
<td>VPOOL SIZE MAX : 0</td>
</tr>
</tbody>
</table>
IFCID 201 - Alter Buffer Pool

The size of the old virtual pool.
Field Name: QW0201OP

VPOOL SIZE (NEW)

The size of the new virtual buffer pool.
Field Name: QW0201NP

VPOOL SEQ THRESH
The old and new virtual pool sequential steal threshold.
Old status taken from the DB2 field QW0201OT.
New status taken from the DB2 field QW0201NT.
Field Name: RT0201VS

VPOOL DWT THRESH
The old and new virtual pool deferred write threshold (DWT).
Old status taken from the DB2 field QW0201OD.
New status taken from the DB2 field QW0201ND.
Field Name: RT0201VD

VPOOL VDWT THRESH - PERCENTAGE
The vertical deferred write threshold for the virtual buffer pool expressed as percentage.
Old status taken from the DB2 field QW0201OV.
New status taken from the DB2 field QW0201NV.
Field Name: RT0201PC

VPOOL VDWT THRESH - BUFFERS
The vertical deferred write threshold for the virtual buffer pool expressed as an absolute number of buffers. It is only used if VERTICAL DEFERRED WRITE THRESHOLD (PERCENTAGE) is 0.
Old status taken from the DB2 field QW0201OJ.
New status taken from the DB2 field QW0201NJ.
Field Name: RT0201BU

VPOOL PLL SEQ THRESH
The old and new virtual pool parallel sequential threshold.
Old status taken from the DB2 field QW0201OQ.
New status taken from the DB2 field QW0201NQ.
Field Name: RT0201VP

ASSISTANT SEQ THRESH
The assisting parallel sequential threshold before and after the ALTER BUFFERPOOL command was issued.
Old status taken from the DB2 field QW0201OX.
New status taken from the DB2 field QW0201NX.
Field Name: RT0201AS
PAGE STEAL METHOD

Identifies the page stealing algorithm (PGSTEAL) that is used for the virtual buffer pool. It controls when and whether performance-critical objects in buffer pools are removed from buffer pools when the space is needed by other objects. Possible values are:

- **LRU** Least recently used (LRU) objects are removed first. This means it takes away pages that are not used so that more recently used pages can remain in the virtual buffer pool. This is used by default.

- **FIFO** First-In-First-Out (FIFO) means that the oldest objects are removed first. This results in a small decrease in the cost of a Getpage operation. It can reduce internal DB2 latch contention in environments that require very high concurrency.

- **NONE**
  Objects are not removed from buffer pool (no page stealing). This setting provides the highest availability for business-critical objects.

Old status taken from the DB2 field QW0201OK.
New status taken from the DB2 field QW0201NK.

**Field Name:** RT0201PS

AUTOSIZE

The old and new status of the AUTOSIZE attribute.
Old status taken from the DB2 field QW0201OZ.
New status taken from the DB2 field QW0201NZ.

**Field Name:** RT0201AT

FRAMESIZE (OLD)

The size of the old frame (4 KB, 1 MB, or 2 GB).

**Field Name:** QW0201OC

FRAMESIZE (NEW)

The new frame size (4 KB, 1 MB, or 2 GB).

**Field Name:** QW0201NC

VPOOL SIZE MIN (OLD)

The minimum size of the old virtual pool.

**Field Name:** QW0201OA

VPOOL SIZE MIN (NEW)

The minimum size of the new virtual pool.

**Field Name:** QW0201NA

VPOOL SIZE MAX (OLD)

The maximum size of the old virtual pool.

**Field Name:** QW0201OB

VPOOL SIZE MAX (NEW)

The maximum size of the new virtual pool.

**Field Name:** QW0201NB
IFCID 202 - Buffer Pool Attributes

This topic shows detailed information about “Record Trace - IFCID 202 - Buffer Pool Attributes”.

Record trace - IFCID 202 - Buffer Pool Attributes

The field labels shown in the following sample layout of “Record Trace - IFCID 202 - Buffer Pool Attributes” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BUFFERPOOL ID</td>
<td>Buffer pool name.</td>
</tr>
<tr>
<td>Field Name: QDBPNM</td>
<td></td>
</tr>
<tr>
<td>VPOOL SIZE</td>
<td>The size of the virtual buffer pool.</td>
</tr>
<tr>
<td>Old status taken from the DB2 field QW0201OP.</td>
<td></td>
</tr>
<tr>
<td>New status taken from the DB2 field QW0201NP.</td>
<td></td>
</tr>
<tr>
<td>Field Name: QDBPVPSZ</td>
<td></td>
</tr>
<tr>
<td>VPOOL VDWT THRESH BUF</td>
<td>The vertical deferred write threshold (VDWQT), shown as the number of</td>
</tr>
<tr>
<td>buffers in the virtual buffer pool that might be occupied by updated pages</td>
<td></td>
</tr>
<tr>
<td>from a single page set.</td>
<td></td>
</tr>
<tr>
<td>Field Name: QDBPVDQB</td>
<td></td>
</tr>
<tr>
<td>PSTEAL METHOD</td>
<td>Identifies the page stealing algorithm (PGSTEAL) that is used for the</td>
</tr>
<tr>
<td>virtual buffer pool. It controls when and whether performance-critical</td>
<td></td>
</tr>
<tr>
<td>objects in buffer pools are removed from buffer pools when the space is</td>
<td></td>
</tr>
<tr>
<td>needed by other objects. Possible values are:</td>
<td></td>
</tr>
<tr>
<td>LRU</td>
<td>Least recently used (LRU) objects are removed first. This means it</td>
</tr>
<tr>
<td>takes away pages that are not used so that more recently used</td>
<td></td>
</tr>
<tr>
<td>pages can remain in the virtual buffer pool. This is used by default.</td>
<td></td>
</tr>
<tr>
<td>FIFO</td>
<td>First-In-First-Out (FIFO) means that the oldest objects are removed</td>
</tr>
<tr>
<td>first. This results in a small decrease in the cost of a Getpage</td>
<td></td>
</tr>
<tr>
<td>operation. It can reduce internal DB2 latch contention in</td>
<td></td>
</tr>
<tr>
<td>environments that require very high concurrency.</td>
<td></td>
</tr>
<tr>
<td>NONE</td>
<td>Objects are not removed from buffer pool (no page stealing). This setting</td>
</tr>
<tr>
<td>provides the highest availability for business-critical objects.</td>
<td></td>
</tr>
<tr>
<td>Field Name: QDBPPGST</td>
<td></td>
</tr>
<tr>
<td>VPOOL SEQ THRESH</td>
<td>Virtual pool sequential threshold (VPSEQT). This threshold is a percentage</td>
</tr>
<tr>
<td>of the virtual buffer pool that might be occupied by sequentially</td>
<td></td>
</tr>
<tr>
<td>accessed pages.</td>
<td></td>
</tr>
<tr>
<td>Field Name: QDBPSEQ</td>
<td></td>
</tr>
</tbody>
</table>

QDBPSLA : 49879
IFCID 202 - Buffer Pool Attributes

pages. The pages can be in the state updated, in use, or available. Therefore, each page might count regarding exceeding any other buffer pool threshold.

The default value for VPSEQT is 80%. You can change this value to a value from 0% to 100% by using the VPSEQT option of the ALTER BUFFERPOOL command.

VPSEQT is checked before stealing a buffer for a sequentially accessed page instead of accessing the page in the virtual buffer pool. If the threshold is exceeded, DB2 tries to steal a buffer that holds a sequentially accessed page rather than one that holds a randomly accessed page.

If you set VPSEQT to 0%, sequential pages cannot occupy space in the virtual buffer pool. In this case, prefetch is disabled, and sequentially accessed pages are discarded when they are released. If you set VPSEQT to 0%, the value of HPSEQT is meaningless because sequential pages that are not kept in the virtual buffer pool do not go in the hiperpool. You can, however, set the value for HPSEQT to a value above zero and the value for VPSEQT to zero. If you set VPSEQT to 100%, sequential pages can monopolize the entire virtual buffer pool.

Field Name: QDBPVPSH

**VPOOL VDWT THRESH (%)**

Vertical deferred write threshold (VDWQT). This threshold is similar to the deferred write threshold but it applies to the number of updated pages for one single page set in the buffer pool. If the percentage or number of updated pages for the data set exceeds the threshold, writes up to 128 pages are scheduled for that data set.

VDWQT can be specified in one of the following ways:

- As a percentage of the virtual buffer pool that might be occupied by updated pages from one single page set. The default value for this threshold is 10%. You can change the percentage to any value from 0% to 90%.
- As the total number of buffers in the virtual buffer pool that might be occupied by updated pages from one single page set. You can specify the number of buffers from 0 to 9999. If you want to use the number of buffers as your threshold, you must set the percentage threshold to 0.

Field Name: QDBPVDQT

**PGFIX ATTRIB**

Indicates whether a page is fixed in real storage when it is first used. It can have one of the following values: YES or NO.

Field Name: QDBPPFIX

**PARALLEL SEQ THRESH**

Virtual buffer pool parallel sequential threshold (VPPSEQT). This threshold is a part of the virtual buffer pool that might support parallel operations. It is measured as a percentage of the sequential steal threshold (VPSEQT). Setting VPPSEQT to zero disables parallel operation.

The default value for this threshold is 50% of the sequential steal threshold (VPSEQT). You can change the default value to any value from 0% to 100% by using the VPPSEQT option on the ALTER BUFFERPOOL command.

Field Name: QDBPPSQT
**VPOOL DWT THRESH**

This threshold is a percentage of the virtual buffer pool that might be occupied by unavailable pages, including updated pages and pages in use.

The default value for QWQT is 50%. You can change this value to any value from 0% to 90% using the DWQT option of the ALTER BUFFERPOOL command.

DB2 checks QWQT when an update to a page is complete. If the percentage of unavailable pages in the virtual buffer pool exceeds QWQT, write operations are scheduled for up to 128 pages per data set to decrease the number of unavailable buffers to 10% below QWQT. For example, if QWQT is 50%, the number of unavailable buffers is reduced to 40%.

When the limit of QWQT is reached, data sets containing the oldest updated pages are written asynchronously. DB2 continues to write pages until the ratio goes below the QWQT.

**Field Name:** QDBPDWQT

**AUTOSIZE**

Indicates if the AUTOSIZE option is activated on the ALTER BUFFERPOOL command.

**Field Name:** QDBPASIZ

**ASS PAR SEQ THRESH**

Virtual buffer pool assisting parallel sequential threshold (VPXPSEQT). This threshold is a part of the virtual buffer pool that might support parallel operations initiated from another DB2 in the data sharing group. It is measured as a percentage of VPPSEQT.

Setting VPXPSEQT to zero (default) prevents DB2 from supporting sysplex query parallelism at run time for queries that use this buffer pool.

You can change the default value to any value from 0% to 100% using the VPXPSEQT option of the ALTER BUFFERPOOL command.

**Field Name:** QDBPXSEQT

**FRAMESIZE**

The frame size.

**Field Name:** QDBPFRAM

**VPOOL SIZE MIN**

The minimum size of the virtual pool.

**Field Name:** QDBPVPMI

**VPOOL SIZE MAX**

The maximum size of the virtual pool.

**Field Name:** QDBPVPMA

**QDBPSLA (Prior to DB2 11)**

This field is for IBM service.

**Field Name:** QDBPSLA
This topic shows detailed information about “Record Trace - IFCID 203 - DDF Heuristic COMMIT/ROLLBK”.

This record reports a heuristic decision that has forced a COMMIT or ROLLBACK for a distributed indoubt thread. The record is produced when a RECOVER INDOUBT command is issued and a remote participant in a distributed thread reports a heuristic rollback or commit during the resynchronization process.

Record trace - IFCID 203 - DDF Heuristic COMMIT/ROLLBK

The field labels shown in the following sample layout of “Record Trace - IFCID 203 - DDF Heuristic COMMIT/ROLLBK” are described in the following section.

DECISION SOURCE

The source of the decision.

Field Name: QW0203LR

DECISION REPORTED

The decision that was reported.

Field Name: QW0203CA

REMOTE DECISION LOCATION

The location, LU name, or IP address (NNN.NNN.NNN) of the location that sent the decision.

Field Name: QW0203LO

NETID

The NETID portion of logical unit of work ID (LUWID).

Field Name: QW0203NT

LUNAME

The LU name portion of the logical unit of work ID (LUWID).

Field Name: QW0203LU

INSTANCE

The instance number portion of the logical unit of work ID (LUWID).

Field Name: QW0203IN

LUW SEQ

The LUW sequence number (commit count) portion of the logical unit of work ID (LUWID).

Field Name: QW0203CM

URID

The recovery log RBA (URID) for the thread.
IFCID 203 - DDF Heuristic COMMIT/ROLLBK

Field Name: QW0203UR

COORDINATOR LOCATION

The location name, LU name, or IP address (NNN.NNN.NNN) of the coordinator.

Field Name: QW0203CO

PARTICIPANT LOCATIONS

The location name of the participants in this unit of work that were accessed directly by this DB2 subsystem.

Field Name: QW0203PA
This topic shows detailed information about “Record Trace - IFCID 204 - DDF Partner Cold Start”.

This record is written when DB2 tries to reconnect to a remote system that requests a cold start. A cold start means that the remote system has no memory of the work that was in progress when the previous connection failed. This record is only produced when DB2 has memory of threads whose outcome must be resolved.

**Record trace - IFCID 204 - DDF Partner Cold Start**

The field labels shown in the following sample layout of “Record Trace - IFCID 204 - DDF Partner Cold Start” are described in the following section.

**LOCATION**

The location, LU name, or IP address (NNN.NNN.NNN) of the remote partner that had the cold start.

Field Name: QW0204LO

**OLD RECOVERY LOG**

The partner’s recovery log name before the cold start.

Field Name: QW0204OR

**NEW RECOVERY LOG**

The partner’s recovery log name after the cold start.

Field Name: QW0204NR

**NETID**

The NETID portion of the logical unit of work ID (LUWID).

Field Name: QW0204NT

**LUNAME**

The LU name portion of the logical unit of work ID (LUWID).

Field Name: QW0204LU

**INSTANCE**

The instance number portion of the logical unit of work ID (LUWID).

Field Name: QW0204IN

**LUW SEQ**

The LUW sequence number (commit count) portion of the logical unit of work ID (LUWID).

Field Name: QW0204CM

**TOKEN**

The local token representing the logical unit of work ID (LUWID).

Field Name: QW0204TK
IFCID 204 - DDF Partner Cold Start

URID
The recovery log RBA (URID) for the thread.
Field Name: QW0204UR

ROLE
The role of DB2 in the LUW.
Field Name: QW0204RL

STATUS
The status of the local DB2 thread.
Field Name: QW0204TS
IFCID 205 - DDF Warm Start Log Name Error Information

This topic shows the data available for IFCID 205.

"IFCID 205 - As Remembered by DB2" on page 40-620
This topic shows detailed information about “Record Trace - IFCID 205 - As Remembered by DB2”.

"IFCID 205 - As Remembered by Partner" on page 40-621
This topic shows detailed information about “Record Trace - IFCID 205 - As Remembered by Partner”.

"IFCID 205 - DDF Warm Start Log Name Error” on page 40-622
This topic shows detailed information about “Record Trace - IFCID 205 - DDF Warm Start Log Name Error”. 
IFCID 205 - As Remembered by DB2

IFCID 205 - As Remembered by DB2

This topic shows detailed information about “Record Trace - IFCID 205 - As Remembered by DB2”.

Record trace - IFCID 205 - As Remembered by DB2

The field labels shown in the following sample layout of “Record Trace - IFCID 205 - As Remembered by DB2” are described in the following section.

AS REMEMBERED BY DB2

PROTOCOL

The protocol used previously as remembered by DB2.

Field Name: QW0205DP

PS HEADER USE

Indicates how the PS header was previously used as remembered by DB2.

Field Name: QW0205DF

LUNAME EXCHANGE

Indicates whether the LU name of the conversation correlator was exchanged in the sync point protocol previously used as remembered by DB2.

Field Name: QW0205DC
IFCID 205 - As Remembered by Partner

This topic shows detailed information about “Record Trace - IFCID 205 - As Remembered by Partner”.

Record trace - IFCID 205 - As Remembered by Partner

The field labels shown in the following sample layout of “Record Trace - IFCID 205 - As Remembered by Partner” are described in the following section.

AS REMEMBERED BY PARTNER PROTOCOL: PRESUMED NOTHING PS HEADER USE: NONE LUNAME EXCHANGE: NO

PROTOCOL

The protocol used previously as remembered by the partner.

Field Name: QW0205PP

PS HEADER USE

Indicates how the PS header was previously used as remembered by the partner.

Field Name: QW0205PF

LUNAME EXCHANGE

Indicates whether the LU name of the conversation correlator was exchanged in the sync point protocol previously used as remembered by the partner.

Field Name: QW0205PC
This topic shows detailed information about “Record Trace - IFCID 205 - DDF Warm Start Log Name Error”.

This record is written when a remote site uses a recovery log name that is different to the last log name used.

Record trace - IFCID 205 - DDF Warm Start Log Name Error

The field labels shown in the following sample layout of “Record Trace - IFCID 205 - DDF Warm Start Log Name Error” are described in the following section.

LOCATION: USIBMSYSTDB2 OUR RECOVERY LOG : LOG NUMBER 1
OUR LOG AS REMEMBERED : LOG NUMBER 2
PARTNER WARM START LOG: LOG NUMBER 3
PARTNER PREVIOUS LOG : LOG NUMBER 4

LOCATION
The location or LU name of the remote partner that had the warm start.
Field Name: QW0205LO

OUR RECOVERY LOG
The name of the local DB2 subsystem's recovery log.
Field Name: QW0205OR

OUR LOG AS REMEMBERED
The name of the local DB2 subsystem's recovery log as remembered by the partner. This field shows 'BLANK' unless the exchange of log names was initiated by the partner.
Field Name: QW0205NR

PARTNER WARM START LOG
The name of the partner's warm start recovery log.
Field Name: QW0205WR

PARTNER PREVIOUS LOG
The name of the partner's previous recovery log.
Field Name: QW0205PR
Record Trace - IFCID 206 - DDF Protocol Error

The field labels shown in the following sample layout of “Record Trace - IFCID 206 - DDF Protocol Error” are described in the following section.

REMOTE LOCATION

The location name or LU name of the remote partner involved in the protocol error.

Field Name: QW0206LO

LAST OPERATION

Indicates whether the last network operation was a send or receive.

Field Name: QW0206SR

DB2 ROLE

The role of DB2 in the logical unit of work (LUW).

Field Name: QW0206RL

DETECTING SITE

The site which detected the error.

Field Name: QW0206DT

NETID

The NETID portion of the logical unit of work ID (LUWID).

Field Name: QW0206NT

LUNAME

The LU name portion of the logical unit of work ID (LUWID).

Field Name: QW0206LU

INSTANCE
IFCID 206 - DDF Protocol Error

The instance number portion of the logical unit of work ID (LUWID).

Field Name: QW0206IN

COMMIT COUNT

The LUW sequence number (commit count) portion of the logical unit of work ID (LUWID).

Field Name: QW0206CM

TOKEN

The local token representing the logical unit of work ID (LUWID).

Field Name: QW0206TK

URID

The recovery log RBA (URID) for the thread.

Field Name: QW0206UR

SENT

The last message sent by this DB2 site during the compare states exchange.

Field Name: QW0206MS

RCVD

The last message received by this DB2 site during the compare states exchange.

Field Name: QW0206MR

VTAM RPL

The VTAM RPL associated with the last compare states message received during the compare states exchange.

Field Name: QW0206VR

EXT

The VTAM RPL extension which describes the LU 6.2 verb indicators for the last message received.

Field Name: QW0206VX
IFCID 207 - DDF Heuristic Damage

This topic shows detailed information about “Record Trace - IFCID 207 - DDF Heuristic Damage”.

This record reports when heuristic damage is detected during the two-phase commit resynchronization. Heuristic damage occurs when a user forces an indoubt unit of work to commit or roll back and the user’s choice conflicts with the outcome chosen by the coordinator of the unit of work.

Record trace - IFCID 207 - DDF Heuristic Damage

The field labels shown in the following sample layout of “Record Trace - IFCID 207 - DDF Heuristic Damage” are described in the following section.

WHERE OCCURRED: SYD1
LOCAL LOCATION: SYD2
UPSTREAM COORDINATOR: SYD3
CICS/IMS COORDINATOR: 'BLANK'

AFFECTED THREADS:
NETID: NETID LUNAME: LUNAME INSTANCE: X'C905E2E3C1D5' LUW SEQ: 1 TOKEN: X'00000002' URID: X'E4D9C9C44040'

ROLE: BOTH
DAMAGE SITE ACTION: ROLLBACK LOCAL SITE ACTION: COMMIT
UPSTREAM SITE ACTION: NO UPSTREAM SITE
LOCAL SITE RECOVERY LOG: DATA 1
UPSTREAM SITE RECOVERY LOG: DATA 3

WHERE OCCURRED
The location, LU name, or IP address (NNN.NNN.NNN) of the location where heuristic damage occurred.
Field Name: QW0207HN

LOCAL LOCATION
The name of this location (the location writing this IFCID).
Field Name: QW0207TN

UPSTREAM COORDINATOR
The location, LU name, or IP address (NNN.NNN.NNN) of the upstream coordinator of this location. This field shows 'BLANK' if this location has no upstream coordinator.
Field Name: QW0207UN

CICS/IMS COORDINATOR
The connection name of the local CICS or IMS coordinator. This field shows 'BLANK' if no local CICS or IMS coordinator exists.
Field Name: QW0207CO

NETID
The NETID portion of the logical unit of work ID (LUWID).
Field Name: QW0207NT

LUNAME
The LU name portion of the logical unit of work ID (LUWID).
Field Name: QW0207LU

INSTANCE
The instance number portion of the logical unit of work ID (LUWID).
Field Name: QW0207IN
IFCID 207 - DDF Heuristic Damage

LUW SEQ
The LUW sequence number (commit count) portion of the logical unit of work ID (LUWID).

Field Name: QW0207CM

TOKEN
The local token representing the logical unit of work ID (LUWID).

Field Name: QW0207TK

URID
The recovery log RBA (URID) for the thread.

Field Name: QW0207UI

ROLE
The role of DB2 in the LUW.

Field Name: QW0207RL

DAMAGE SITE ACTION
The action taken by the site with the heuristic damage.

Field Name: QW0207HA

LOCAL SITE ACTION
The action taken by the local site.

Field Name: QW0207TA

UPSTREAM SITE ACTION
The action taken by the upstream coordinator if one exists.

Field Name: QW0207UA

DAMAGE SITE RECOVERY LOG
The recovery log name of the site where the heuristic damage occurred.

Field Name: QW0207HR

LOCAL SITE RECOVERY LOG
The recovery log name of the local location.

Field Name: QW0207TR

UPSTREAM SITE RECOVERY LOG
The recovery log name of the upstream coordinator (if an upstream coordinator exists).

Field Name: QW0207UR
IFCID 208 - DDF Syncpoint Protocol Error

Record trace - IFCID 208 - DDF Syncpoint Protocol Error

The field labels shown in the following sample layout of “Record Trace - IFCID 208 - DDF Syncpoint Protocol Error” are described in the following section.

REMOTE LOCATION
The location name or LU name of the remote partner involved in the protocol error.
Field Name: QW0208LO

LAST OPERATION
Indicates whether the last network operation was a send or receive.
Field Name: QW0208SR

DB2 ROLE
The role of DB2 in the LUW.
Field Name: QW0208RL

DETECTING SITE
The site which detected the error.
Field Name: QW0208DT

LOCAL THREAD STATUS
The status of the local DB2 thread.
Field Name: QW0208TS

ASSUMED REMOTE THREAD STATUS
The assumed status of the remote thread.
Field Name: QW0208PS

NETID
The NETID portion of the logical unit of work ID (LUWID).
Field Name: QW0208NT
IFCID 208 - DDF Syncpoint Protocol Error

**LUNAME**
The LU name portion of the logical unit of work ID (LUWID).
*Field Name:* QW0208LU

**INSTANCE**
The instance number portion of the logical unit of work ID (LUWID).
*Field Name:* QW0208IN

**COMMIT COUNT**
The LUW sequence number (commit count) portion of the logical unit of work ID (LUWID).
*Field Name:* QW0208CM

**TOKEN**
The local token representing the logical unit of work ID (LUWID).
*Field Name:* QW0208TK

**URID**
The recovery log RBA (URID) for the thread.
*Field Name:* QW0208UR

**SENT**
The last message sent by this DB2 site during sync point processing.
*Field Name:* QW0208MS

**RCVD**
The last message received by this DB2 site during sync point processing.
*Field Name:* QW0208MR

**VTAM RPL**
The VTAM RPL associated with the last compare states message received during the compare states exchange.
*Field Name:* QW0208VR

**EXT**
The VTAM RPL extension which describes the LU 6.2 verb indicators for the last message received.
*Field Name:* QW0208VX
This topic shows detailed information about “Record Trace - IFCID 209 - DDF Syncpoint Comm Failure”.

This record is written when a communication failure occurs after phase 1 of the SNA commit process. The thread that experiences the communication failure might still be indoubt at the participant location.

Record trace - IFCID 209 - DDF Syncpoint Comm Failure

The field labels shown in the following sample layout of “Record Trace - IFCID 209 - DDF Syncpoint Comm Failure” are described in the following section.

**REMOTE PARTNER LOCATION**

The location, LU name, or IP address (NNN.NNN.NNN) of the remote partner involved in the communication error.

*Field Name:* QW0209LO

**NETID**

The NETID portion of the logical unit of work ID (LUWID).

*Field Name:* QW0209NT

**LUNAME**

The LU name portion of the logical unit of work ID (LUWID).

*Field Name:* QW0209LU

**INSTANCE**

The instance number portion of the logical unit of work ID (LUWID).

*Field Name:* QW0209IN

**LUW SEQ**

The LUW sequence number (commit count) portion of the logical unit of work ID (LUWID).

*Field Name:* QW0209CM

**TOKEN**

The local token representing the logical unit of work ID (LUWID).

*Field Name:* QW0209TK

**URID**

The recovery log RBA (URID) for the thread.

*Field Name:* QW0209UR

**ROLE**

The role of DB2 in the logical unit of work (LUW).

*Field Name:* QW0209RL

**LOCAL THREAD STATUS**
IFCID 209 - DDF Syncpoint Comm Failure

The status of the local DB2 thread.

Field Name: QW0209TS
This topic shows detailed information about “Record Trace - IFCID 210 - Warm Start Log Name Change”.

This record is written when a remote site warm starts with a recovery log name that is different from its previous recovery log name. DB2 has no threads that require resolution, so the new recovery log name is accepted.

**Record trace - IFCID 210 - Warm Start Log Name Change**

The field labels shown in the following sample layout of “Record Trace - IFCID 210 - Warm Start Log Name Change” are described in the following section.

**LOCATION:** SYD2  
**WARM START RECOVERY LOG:** CURRENT RECOVERY LOG  
**PREVIOUS RECOVERY LOG:** PREVIOUS RECOVERY LOG

**LOCATION**

The location, LU name, or IP address (NNN.NNN.NNN) of the remote partner that sent the warm start indication.

*Field Name:* QW0210LO

**WARM START RECOVERY LOG**

The name of the partner’s warm start recovery log.

*Field Name:* QW0210WR

**PREVIOUS RECOVERY LOG**

The name of the partner’s previous recovery log.

*Field Name:* QW0210PR
This topic shows detailed information about “Record Trace - IFCID 211 - Claim Data”.

This record contains information about making and releasing a claim. One record is written for each request to make a claim or release a claim.

**Record trace - IFCID 211 - Claim Data**

The field labels shown in the following sample layout of “Record Trace - IFCID 211 - Claim Data” are described in the following section.

**DBID**

The database identifier of the object of the claim request. This field contains 0 if the request is for a release of all claims.

*Field Name:* QW0211DB

**PSID**

The page set identifier of the object of the claim request. This field contains 0 if the request is for a release of all claims.

*Field Name:* QW0211PS

**PARTITION NO.**

The partition number of the object of the claim request. This field contains 0 if the request is for a release of all claims or if the table space or index space is not partitioned (and the claim request is at the page set level rather than the logical partition level).

*Field Name:* QW0211PT

**CLAIM REQUEST TYPE**

The claim request type.

*Field Name:* QW0211RQ

**CLAIM CLASS**

The claim class.

*Field Name:* QW0211CC

**CLAIM DURATION**

The claim duration. This field shows 'BLANK' if the claim is released.

*Field Name:* QW0211DU

**CLAIM RESULT**

The result of the claim request.

*Field Name:* QW0211RC

**REASON IF CLAIM UNSUCCESSFUL**

The reason for an unsuccessful claim. This field is only printed if the value in CLAIM RESULT is UNSUCCESSFUL.

*Field Name:* QW0211RS
This topic shows detailed information about “Record Trace - IFCID 212 - Drain Data”.

This record contains information about requesting and releasing a drain or a pseudo drain. One record is written for each drain or release request on a claim class. Another record is written for a drain that is only waiting for the claimers to release claims and not acquiring a drain lock (pseudo drain).

Record trace - IFCID 212 - Drain Data

The field labels shown in the following sample layout of “Record Trace - IFCID 212 - Drain Data” are described in the following section.

<table>
<thead>
<tr>
<th>DBID</th>
<th>CATD3DB1</th>
<th>PSID</th>
<th>CATD3TS2</th>
<th>PARTITION NO.</th>
<th>2</th>
<th>DRAIN REQUEST TYPE</th>
<th>DRAIN LOCK MODE</th>
<th>CLAIM CLASS</th>
<th>WRITE</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRAIN LOCK MODE</td>
<td>EXCLUSIVE</td>
<td>DRAIN RESULT</td>
<td>SUCCESSFUL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

DBID

The database identifier of the object of the drain request. This field contains 0 if the request is for a release of all drains.

Field Name: QW0212DB

PSID

The page set identifier of the object of the drain request. This field contains 0 if the request is for a release of all drains.

Field Name: QW0212PS

PARTITION NO.

The partition number of the object of the drain request. This field contains 0 if the request is for a release of all drains or if the table space or index space is non-partitioned (and the drain request is at the page set level rather than the logical partition level).

Field Name: QW0212PT

DRAIN REQUEST TYPE

The drain request type.

Field Name: QW0212RQ

CLAIM CLASS

The claim class.

Field Name: QW0212CC

DRAIN LOCK MODE

The mode of the drain lock requested. This field shows ‘BLANK’ if the drain is released or no lock is requested.

Field Name: QW0212MO

DRAIN RESULT

The result of the drain request.

Field Name: QW0212RC

REASON IF DRAIN UNSUCCESSFUL
IFCID 212 - Drain Data

The reason for an unsuccessful drain. This field is only printed if the value in CLAIM RESULT is UNSUCCESSFUL.

Field Name: QW0212RS
IFCID 213 - Drain Lock Wait Start

This topic shows detailed information about “Record Trace - IFCID 213 - Drain Lock Wait Start”.

This record contains information about the beginning of a wait for a drain lock. For drain locks, this record is written instead of IFCID 44.

Record trace - IFCID 213 - Drain Lock Wait Start

The field labels shown in the following sample layout of “Record Trace - IFCID 213 - Drain Lock Wait Start” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOCK HASH VALUE</td>
<td>The hash value of the locked resource.</td>
</tr>
<tr>
<td>LOCK NAME LENGTH</td>
<td>The length of the lock name.</td>
</tr>
<tr>
<td>LOCK QUALIFIER</td>
<td>The lock qualifier.</td>
</tr>
<tr>
<td>LOCK RES TYPE</td>
<td>The locked resource type or the type of locking operation.</td>
</tr>
<tr>
<td>DBID</td>
<td>The database ID of the object of the claim request.</td>
</tr>
<tr>
<td>PSID</td>
<td>The page set identifier of the object of the claim request.</td>
</tr>
<tr>
<td>PARTITION NO.</td>
<td>The partition number of the object of the lock request. This field contains 0 if the table space or index space is not partitioned (and the lock request is at the page set level rather than the logical partition level).</td>
</tr>
<tr>
<td>IRLM FUNCTION</td>
<td>The IRLM function.</td>
</tr>
<tr>
<td>STATE</td>
<td></td>
</tr>
</tbody>
</table>
IFCID 213 - Drain Lock Wait Start

The lock state.

Field Name: QW0213ST

REASON SUST

The reason for the suspension.

Field Name: QW0213WS
IFCID 214 - Drain Lock Wait End

This topic shows detailed information about “Record Trace - IFCID 214 - Drain Lock Wait End”.

This record contains information about the end of a wait for a drain lock. For drain locks, this record is written instead of IFCID 45.

Record trace - IFCID 214 - Drain Lock Wait End

The field labels shown in the following sample layout of “Record Trace - IFCID 214 - Drain Lock Wait End” are described in the following section.

DRAIN LOCK <-- NETWORKID: DEIBMIPS LUNAME: IPSAQ811 LUWSEQ: 1
WAIT END  REASON FOR RESUME : NORMAL RESUME
REASON FOR SUSPEND : X'80'
IRLM LATCH CONTENTION : YES
IRLM QUEUED REQUEST : NO
LOCAL RESOURCE CONTENTION : NO
GLOBAL RESOURCE CONTENTION : NO
INTER-SYSTEM MESSAGE SENDING : NO
GLOBAL CONTENTION EXTENT : X'20'
XES GLOBAL CONTENTION : NO
IRLM GLOBAL CONTENTION : NO
FALSE CONTENTION : NO
QW0214W4 NO QW0214W6 NO QW0214W8 NO
QW0214X1 NO QW0214X2 NO QW0214X5 NO
QW0214X6 NO QW0214X7 NO QW0214X8 NO

REASON FOR RESUME
The reason for the lock resume.
Field Name: QW0214R

REASON FOR SUSPEND
The reason for the suspension. The nonserviceability values are:
Field Name: QW0214SR

IRLM LATCH CONTENTION
Indicates whether IRLM latch contention occurred.
Field Name: QW0214W1

IRLM QUEUED REQUEST
Indicates whether IRLM queued request occurred.
Field Name: QW0214W2

LOCAL RESOURCE CONTENTION
Indicates whether local resource contention occurred.
Field Name: QW0214W3

GLOBAL RESOURCE CONTENTION
Indicates whether intersystem communication was required to resolve an IRLM request.
Field Name: QW0214W5

INTER-SYSTEM MESSAGE SENDING
Indicates whether any intersystem messages were sent.

Field Name: QW0214W7

GLOBAL CONTENTION EXTENT

The extent of global contention. This is applicable only if the value in GLOBAL RESOURCE CONTENTION is YES. The nonserviceability values are:

Field Name: QW0214XR

XES GLOBAL CONTENTION

Indicates whether XES global resource contention occurred.

Field Name: QW0214X3

IRLM GLOBAL CONTENTION

Indicates whether IRLM global resource contention occurred.
Indicates whether there was IRLM or XES global resource contention.

Field Name: QW0214X4
IFCID 215 - Claim Count 0 Wait Start

This topic shows detailed information about “Record Trace - IFCID 215 - Claim Count 0 Wait Start”.

This IFCID records the beginning of a wait for the number of pending claims to reach 0.

Record trace - IFCID 215 - Claim Count 0 Wait Start

The field labels shown in the following sample layout of “Record Trace - IFCID 215 - Claim Count 0 Wait Start” are described in the following section.

CLAIM CNT 0---> 'BLANK'
WAIT START NETWORKID: DEIBMIPS LUNAME: IPSAQ811 LWSEQ: 1
DBID: 1 PSID: 2 PARTITION NO. 3
CLAIM CLASS: RR READ CLAIM COUNT: 20

DBID
The database identifier of the object of the drain request.
Field Name: QW0215DB

PSID
The page set identifier of the object of the drain request.
Field Name: QW0215PS

PARTITION NO.
The partition number of the object of the drain request. This field contains 0 if the object is a non-partitioned table space or non-partitioned index being drained at the page set level.
Field Name: QW0215PT

CLAIM CLASS
The claim class.
Field Name: QW0215CC

CLAIM COUNT
The number of claims pending for this resource.
Field Name: QW0215CT
This topic shows detailed information about “Record Trace - IFCID 216 - Claim Count 0 Wait End”.

This IFCID records the end of a wait for a claim count to reach 0.

**Record trace - IFCID 216 - Claim Count 0 Wait End**

The field labels shown in the following sample layout of “Record Trace - IFCID 216 - Claim Count 0 Wait End” are described in the following section.

```
CLAIM CNT 0<-- 'BLANK'
WAIT END NETWORKID: DEIBMIPS LUNAME: IPSAQ811 LUWSEQ: 1
DBID: 1 PSID: 2 PARTITION NO. 3
CLAIM CLASS: RR READ REASON FOR RESUME: TIMEOUT
```

**DBID**

The database identifier of the object of the drain request.

**Field Name:** QW0216DB

**PSID**

The page set identifier of the object of the drain request.

**Field Name:** QW0216PS

**PARTITION NO.**

The partition number of the object of the drain request. This field contains 0 if the object is a non-partitioned table space or non-partitioned index being drained at the page set level.

**Field Name:** QW0216PT

**CLAIM CLASS**

The claim class.

**Field Name:** QW0216CC

**REASON FOR RESUME**

The reason for the resume.

**Field Name:** QW0216R
IFCID 217 - Storage Pools

This record only contains data for IFCID 217.

“IFCID 217 - Additional Information” on page 40-642
This topic shows detailed information about “Record Trace - IFCID 217 - Additional Information”.

“IFCID 217 - Agent Local Storage Pool Sizes” on page 40-643
This topic shows detailed information about “Record Trace - IFCID 217 - Agent Local Storage Pool Sizes”.

“IFCID 217 - DBM1 Storage Pool Sizes” on page 40-646
This topic shows detailed information about “Record Trace - IFCID 217 - DBM1 Storage Pool Sizes”.

“IFCID 217 - Storage Manager Pool Statistics” on page 40-648
This topic shows detailed information about “Record Trace - IFCID 217 - Storage Manager Pool Statistics”.

Chapter 40. IFCID Record Blocks  40-641
IFCID 217 - Additional Information

IFCID 217 - Additional Information
This topic shows detailed information about “Record Trace - IFCID 217 - Additional Information”.

Record trace - IFCID 217 - Additional Information

The field labels shown in the following sample layout of “Record Trace - IFCID 217 - Additional Information” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TOTAL DICTIONARY STORAGE</td>
<td>Storage space allocated for the compression dictionary.</td>
</tr>
<tr>
<td>Field Name: QW02174D</td>
<td></td>
</tr>
<tr>
<td>TOTAL STORAGE IN AGENT LOCAL POOLS</td>
<td>Storage used by system agents.</td>
</tr>
<tr>
<td>Field Name: QW02174T</td>
<td></td>
</tr>
<tr>
<td>NO ACTIVE ALLIED THREADS</td>
<td>The number of active allied threads.</td>
</tr>
<tr>
<td>Field Name: QW02174A</td>
<td></td>
</tr>
<tr>
<td>NO CASTOUT ENGINES</td>
<td>Number of engines available for data-sharing castout processing.</td>
</tr>
<tr>
<td>Field Name: QW02174C</td>
<td></td>
</tr>
<tr>
<td>NO P-LOCK/NOTIFY EXIT ENGINES</td>
<td>Number of data sharing P-Lock engines and Notify Exit engines.</td>
</tr>
<tr>
<td>Field Name: QW02174E</td>
<td></td>
</tr>
<tr>
<td>NO PREFETCH ENGINES</td>
<td>Number of engines used for sequential, list, and dynamic prefetch.</td>
</tr>
<tr>
<td>Field Name: QW02174F</td>
<td></td>
</tr>
<tr>
<td>NO GBP WRITE ENGINES</td>
<td>Number of engines for group buffer pool writes.</td>
</tr>
<tr>
<td>Field Name: QW02174G</td>
<td></td>
</tr>
<tr>
<td>NO DEFERRED WRITE ENGINES</td>
<td>Number of engines used for deferred write operations.</td>
</tr>
<tr>
<td>Field Name: QW02174W</td>
<td></td>
</tr>
</tbody>
</table>
IFCID 217 - Agent Local Storage Pool Sizes

Record trace - IFCID 217 - Agent Local Storage Pool Sizes

The field labels shown in the following sample layout of “Record Trace - IFCID 217 - Agent Local Storage Pool Sizes” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TOTAL POOL STORAGE</td>
<td>Total storage in the agent local pool.</td>
</tr>
<tr>
<td>STORAGE CLASS</td>
<td>Storage class for agent local pools.</td>
</tr>
<tr>
<td>CONNECTION NAME</td>
<td>The connection name. Possible values are:</td>
</tr>
<tr>
<td></td>
<td>• For batch: BATCH</td>
</tr>
<tr>
<td></td>
<td>• For TSO: TSO</td>
</tr>
<tr>
<td></td>
<td>• For QMF: DB2CALL</td>
</tr>
<tr>
<td></td>
<td>• For utilities: UTILITY</td>
</tr>
<tr>
<td></td>
<td>• For DB2 private protocol this is the DB2 subsystem ID</td>
</tr>
<tr>
<td></td>
<td>• For IMS: the IMS ID</td>
</tr>
<tr>
<td></td>
<td>• For CICS, this is the CICS ID</td>
</tr>
<tr>
<td></td>
<td>• For DRDA connections from non-DB2 requesters: SERVER</td>
</tr>
<tr>
<td>CORRELATION ID</td>
<td>Correlation identifier.</td>
</tr>
<tr>
<td>PLAN NAME</td>
<td>The plan name. It is blank for a DB2 command thread; otherwise:</td>
</tr>
<tr>
<td></td>
<td>• DSNESPRR For SPUFI with repeatable read.</td>
</tr>
<tr>
<td></td>
<td>• DSNESPCS For SPUFI with cursor stability.</td>
</tr>
<tr>
<td></td>
<td>• DSNUTIL For utilities.</td>
</tr>
<tr>
<td></td>
<td>• DSNTEP2 For DSNTEP2.</td>
</tr>
</tbody>
</table>

TOTAL POOL STORAGE

Field Name: QW02173T

STORAGE CLASS

Field Name: QW02173L

CONNECTION NAME

Field Name: QW0217QN

CORRELATION ID

Field Name: QW0217QR

PLAN NAME

Field Name: QW0217QR

Chapter 40. IFCID Record Blocks 40-643
IFCID 217 - Agent Local Storage Pool Sizes

DSNBIND
For binding.

The application plan name
For IMS.

The application plan name
For CICS.

A blank plan name
For IMS and CICS commands.

DSQPLAN
For QMF.

The first 8 bytes of the application name
For DRDA connections to the common servers.

Field Name: QW0217QP
WORKSTATION NAME
The end user's workstation name.
Field Name: QW0217QW
TRANSACTION NAME
The transaction or application name that is run.
Field Name: QW0217QX
MVS SUBPOOL
MVS subpool.
Field Name: QW0217BP
FIXED STORAGE POOL
Indicates if the storage pool is fixed.
Field Name: QW02173X
VARIA STORAGE POOL
Indicates if the storage pool is variable.
Field Name: QW02173R
OWNING ASID
The amount of storage allocated for agent-related local storage. This storage is used for operations such as sort.

Background and Tuning Information
Sorting requires a large amount of virtual storage because there can be multiple copies of the data being sorted at a given time.

DB2 Sort uses two kinds of storage pool for various internal control structures and data records, an agent-related local storage pool and a global sort pool. To take advantage of the 64-bit addressability for larger storage pool, some high level sort control structures are kept in agent-related storage below the 2 GB bar, which contain 64-bit pointers to areas in the global sort pool above the 2 GB bar. The sort pool above 2 GB contains sort tree nodes and data buffers.

Field Name: QW0217AL
IFCID 217 - Agent Local Storage Pool Sizes

AUTHORIZATION ID

The primary authorization ID from a connection or signon. The connection authorization exit and the signon authorization exit can change the primary authorization ID so that it differs from the original primary authorization ID (ORIGAUTH). Distributed authorization ID translation can also change the primary authorization ID.

Field Name: QW0217QC

USERID

The user ID of the workstation end user. This user ID can be different from the authorization ID used to connect to DB2. This field contains blanks if the client does not supply this information.

Field Name: QW0217QD

QW02173H

This field is for IBM service.

Field Name: QW02173H

QW02173F

Storage pool flags:
• Fixed storage pool
• Variable storage pool
• More agent storage pool data in one or more ifcid 0217 records after this one.
• This is the last ifcid 0217 record in this sequence of agent storage pool data.
• This is a parent task for parallelism.
• This is a child task for parallelism.

Field Name: QW02173F

QW02173C

This field is for IBM service.

Field Name: QW02173C
IFCID 217 - DBM1 Storage Pool Sizes

This topic shows detailed information about “Record Trace - IFCID 217 - DBM1 Storage Pool Sizes”.

Record trace - IFCID 217 - DBM1 Storage Pool Sizes

The field labels shown in the following sample layout of “Record Trace - IFCID 217 - DBM1 Storage Pool Sizes” are described in the following section.

**TOTAL POOL STORAGE**

Total amount of DBM1 storage available for pools.

Field Name: QW0217ST

**STORAGE CLASS**

Storage class.

Field Name: QW0217CL

**MVS SUBPOOL**

MVS subpool.

Field Name: QW0217BP

**DESCRIPTION**

Storage pool description.

Field Name: QW0217DE

**FIXED STORAGE POOL**

Indicates if the storage pool is fixed.

Field Name: QW02173X

**VARIA STORAGE POOL**

Indicates if the storage pool is variable.

Field Name: QW02173R

**OWNING ASID**

The amount of storage allocated for agent-related local storage. This storage is used for operations such as sort.

Background and Tuning Information

Sorting requires a large amount of virtual storage because there can be multiple copies of the data being sorted at a given time.

DB2 Sort uses two kinds of storage pool for various internal control structures and data records, an agent-related local storage pool and a global sort pool. To take advantage of the 64-bit addressability for larger storage pool, some high level sort control structures are kept in agent-related storage below the 2 GB bar, which contain 64-bit pointers to areas in the global sort pool above the 2 GB bar. The sort pool above 2 GB contains sort tree nodes and data buffers.
IFCID 217 - DBM1 Storage Pool Sizes

Field Name: QW0217AL
QW0217PH
Contains QW0217PH.
Field Name: QW0217PH
QW0217FL
Storage pool flags.
   Fixed storage pool
   Variable storage pool
More DBM1 local data follows, in one or more ifcid 0217 records after this one.
This is the last ifcid 0217 record in this sequence of local DBM1 data. ifcid 0217 records with agent local storage pool data follow.
Field Name: QW0217FL
IFCID 217 - Storage Manager Pool Statistics

This topic shows detailed information about “Record Trace - IFCID 217 - Storage Manager Pool Statistics”.

Record trace - IFCID 217 - Storage Manager Pool Statistics

The field labels shown in the following sample layout of “Record Trace - IFCID 217 - Storage Manager Pool Statistics” are described in the following section.

### STORAGE MANAGER POOL STATISTICS

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TOTAL AVAILABLE STORAGE</td>
<td>The total amount of storage available for storage manager pools. Field Name: QW0217AV</td>
</tr>
<tr>
<td>AMOUNT FOR MVS USE</td>
<td>The amount of storage available for operating system activity. Field Name: QW0217MV</td>
</tr>
<tr>
<td>RESERVED FOR MUST-COMPLETE</td>
<td>The storage cushion warning to contract. Field Name: QW0217SO</td>
</tr>
<tr>
<td>TOTAL GETMAIN STORAGE BELOW 2GB</td>
<td>Total storage acquired by GETMAIN. This includes space for virtual pools, EDM pool, compression dictionary, castout buffers, and the data space lookaside buffer, hiperpool control blocks, and data space buffer pool control blocks. This figure can be different from the sum of GETMAIN storage items shown in the statistics DBM1 storage. Field Name: QW0217GM</td>
</tr>
<tr>
<td>TOTAL GETMAIN STACK STORAGE</td>
<td>The amount of storage allocated for agent-related local storage. This storage is used for operations such as sort. Field Name: QW0217CM</td>
</tr>
</tbody>
</table>

**Background and Tuning Information**

Sorting requires a large amount of virtual storage because there can be multiple copies of the data being sorted at a given time.
DB2 Sort uses two kinds of storage pool for various internal control structures and data records, an agent-related local storage pool and a global sort pool. To take advantage of the 64-bit addressability for larger storage pool, some high level sort control structures are kept in agent-related storage below the 2 GB bar, which contain 64-bit pointers to areas in the global sort pool above the 2 GB bar. The sort pool above 2 GB contains sort tree nodes and data buffers.

**Field Name:** QW0217AL

**GETMAINED STORAGE ABOVE 2GB**

The storage above 2 GB acquired by GETMAIN.

**Field Name:** QW0217GA

**DATA ABOVE 2GB IS INCOMPLETE**

Data above 2 GB is incomplete. It can be one of the following:
- YES
- NO

**Field Name:** QW0217HF

**TOTAL VIRTUAL SHRED ABOVE BAR**

Total amount of DBM1 storage available for pools.

**Field Name:** QW0217ST

**24 BIT LOW PRIVATE**

The amount of private MVS storage below the 16 MB line. This storage is obtained from bottom upward, usually for unauthorized programs.

**Field Name:** QW0217LO

**24 BIT HIGH PRIVATE**

The amount of private MVS storage below the 16 MB line. This storage is obtained from top downward, usually for authorized programs.

**Field Name:** QW0217HI

**31 BIT EXTENDED LOW PRIVATE**

The amount of private MVS storage above the 16 MB line. This storage is obtained from bottom upward, usually for unauthorized programs.

**Field Name:** QW0217EL

**31 BIT EXTENDED HIGH PRIVATE**

The amount of private MVS storage above the 16 MB line. This storage is obtained from top downward, usually for authorized programs.

**Field Name:** QW0217EH

**EXTENDED REGION SIZE (MAX)**

The maximum amount of MVS private storage available above the 16 MB line.

**Field Name:** QW0217RG

**EXTENDED CSA SIZE**

The size of the common storage area (CSA) above the 16 MB line.
IFCID 217 - Storage Manager Pool Statistics

Field Name: QW0217EC

DB2 V10 TOKEN

The size of the common storage area (CSA) above the 16 MB line.

Field Name: QW0217TK
## IFCID 218 - Lock Avoidance Summary

This topic shows detailed information about “Record Trace - IFCID 218 - Lock Avoidance Summary”.

This record indicates whether a successful lock avoidance test occurred during a given unit of work. The record is externalized at the agent at each commit or rollback.

### Record trace - IFCID 218 - Lock Avoidance Summary

The field labels shown in the following sample layout of “Record Trace - IFCID 218 - Lock Avoidance Summary” are described in the following section.

**LOCK AVOID DURING UNIT OF WORK**: YES NO

- **NO. PAGE SET SUBRECORDS:** 4
- **DBID:** INSQDB  PSID: EVENTS  LOCK AVOID DURING UNIT OF WORK: YES
- **DBID:** DSNDB06  PSID: SYSDBAUT  LOCK AVOID DURING UNIT OF WORK: NO

**LOCK AVOID DURING UNIT OF WORK**

Indicates whether there was a successful lock avoidance test during this unit of work.

**Field Name:** QW0218CT

**NO. PAGE SET SUBRECORDS**

The number of page set subrecords contained in this record. The fields DBID, PSID, and LOCK AVOID DURING UNIT OF WORK are repeated for each page set that has a lock avoidance test.

**Field Name:** QW0218N

**DBID**

The database ID.

**Field Name:** QW0218PD

**PSID**

The page set ID.

**Field Name:** QW0218PP

**LOCK AVOID DURING UNIT OF WORK**

Indicates whether there was a successful lock avoidance test for this page set during this unit of work.

**Field Name:** QW0218PC
IFCID 219 - Utility LISTDEF List Information

This topic shows detailed information about “Record Trace - IFCID 219 - Utility LISTDEF List Information”.

Record trace - IFCID 219 - Utility LISTDEF List Information

The field labels shown in the following sample layout of “Record Trace - IFCID 219 - Utility LISTDEF List Information” are described in the following section.

LIST NAME

Name of list definition information.

Field Name: QW0219LN

LIST TYPE

Type of LISTDEF information:

T Table space list.

I Index space list.

M Mixed list.

Field Name: QW0219LT

LIST SIZE

Number of entries in the LISTDEF.

Field Name: QW0219LS
IFCID 220 - Utility Data Set Information

This topic shows detailed information about “Record Trace - IFCID 220 - Utility Data Set Information”.

This record is written when a data set is closed.

Record trace - IFCID 220 - Utility Data Set Information

The field labels shown in the following sample layout of “Record Trace - IFCID 220 - Utility Data Set Information” are described in the following section.

DD NAME: LOCATION DATASET NAME: MYDATASET.NAME.CAN.BE.UP.TO.FORTY.FOUR.CHARS TEMPLATE NAME: MYTEMPLA
OPEN TIME STAMP: 14/09/00;5 DEVICE TYPE: D

DD NAME
  Data definition.
  Field Name: QW0220DD

DATA SET NAME
  Data set name.
  Field Name: QW0220DN

TEMPLATE NAME
  Template name.
  Field Name: QW0220TN

NO. READS
  Number of READ operations.
  Field Name: QW0220RD

NO. WRITES
  Number of WRITE operations.
  Field Name: QW0220WR

NO. CHECKS
  Number of checks.
  Field Name: QW0220CH

NO. EOVS
  Number of End of Volumes.
  Field Name: QW0220EV

I/O WAIT TIME
  I/O wait time in milliseconds.
  Field Name: QW0220WT

OPEN TIME STAMP
  Time the data set was opened.
  Field Name: QW0220OT

DEVICE TYPE
Device type:
D     DASD.
T     Tape.

Field Name: QW0220DT
IFCID 221 - Parallel Group Execution

This topic shows the data available for IFCID 221.

“IFCID 221 - Buffer Pool Constrained Data (Section Type C)” on page 40-656
This topic shows detailed information about “Record Trace - IFCID 221 - Buffer Pool Constrained Data (Section Type C)”.

“IFCID 221 - Detail Buffer Pool Constrained Data (Section Type E)” on page 40-657
This topic shows detailed information about “Record Trace - IFCID 221 - Detail Buffer Pool Constrained Data (Section Type E)”.

“IFCID 221 - Parallel Data” on page 40-658
This topic shows detailed information about “Record Trace - IFCID 221 - Parallel Data”.

“IFCID 221 - Section Type D” on page 40-661
This topic shows detailed information about “Record Trace - IFCID 221 - Section Type D”.

Chapter 40. IFCID Record Blocks 40-655
**IFCID 221 - Buffer Pool Constrained Data (Section Type C)**

This topic shows detailed information about “Record Trace - IFCID 221 - Buffer Pool Constrained Data (Section Type C)”.

**Record trace - IFCID 221 - Buffer Pool Constrained Data (Section Type C)**

The field labels shown in the following sample layout of “Record Trace - IFCID 221 - Buffer Pool Constrained Data (Section Type C)” are described in the following section.

```
BUFFERPOOL CONSTRAINED DATA
LENGTH: X'0010'
DBID PSID TYPE BPID WITH_SECT.E
0 0 W 7 0
0 0 W 7 0
```

**LENGTH**

The total length of all entries.

Field Name: QW0221CL

**DBID**

The database identifier.

Field Name: QW0221DB

**PSID**

The page set identifier.

Field Name: QW0221PS

**TYPE**

The type of page set:

- T Table space
- I Index
- W Work file

Field Name: QW0221TY

**BPID**

The buffer pool identifier.

Field Name: QW0221BP

**WITH_SECT.E**

The number of detail buffer pool constrained data sections to follow in section type E.

Field Name: QW0221DN
IFCID 221 - Detail Buffer Pool Constrained Data (Section Type E)

This topic shows detailed information about “Record Trace - IFCID 221 - Detail Buffer Pool Constrained Data (Section Type E)”.

LENGTH
   The total length of all entries.
   Field Name: QW0221CL

DB2_MEMBER
   The name of the DB2 member.
   Field Name: QW0221MN

CONSTRANGED
   Indicates whether the DB2 member is constrained.
   Field Name: QW0221CS
IFCID 221 - Parallel Data

IFCID 221 - Parallel Data
This topic shows detailed information about “Record Trace - IFCID 221 - Parallel Data”.

Record trace - IFCID 221 - Parallel Data
The field labels shown in the following sample layout of “Record Trace - IFCID 221 - Parallel Data” are described in the following section.

PARALLEL DATA
LOCATION NAME : 'BLANK'
PKG COLLECTION ID : 'BLANK'
PROGRAM NAME : 'BLANK'
STMT.NO : 1077952576 QUERYBLOCK NUMBER : 16448 CONS.TOKEN : X'4040404040404040'
PLANNED(BIND) DEGREE: 16448 REP.SECTION TYPE : N/P PARALL.GROUP NO: 16448
PLANNED(RUN) DEGREE: 16448 ACTUAL(RUN) DEGREE: 16448
TYPE OF PARALLELISM : X'40'
NUMBER OF MEMBERS : 1077952576 REASON : X'40'
H/L PARTITION TYPE: LOGICAL

LOCATION NAME
The location name or RDB name.
Field Name: QW0221LN

PKG COLLECTION ID
The package collection ID.
Field Name: QW0221PC

PROGRAM NAME
The program name.
Field Name: QW0221PN

STMT.NO
The statement number. It is the same as the QUERYNO in the PLAN_TABLE, if the PLAN_TABLE exists.
Field Name: QW0221SN

QUERYBLOCK NUMBER
The query block number. It is the same as the QBLOCKNO in the PLAN_TABLE, if the PLAN_TABLE exists.
Field Name: QW0221QN

CONS.TOKEN
The timestamp (consistency token).
Field Name: QW0221TS

PLANNED(BIND) DEGREE
The planned degree of parallelism at bind time. Parallelism decisions are made at bind time. However, the value in this field is 0 if the statement has host variables, because host variables cause the parallelism decision to be made at run time. See field PLANNED(RUN) DEGREE.
Field Name: QW0221PD

REP.SECTION TYPE
The type of the repeating section.
Field Name: QW0221TP
PARALLEL GROUP NO
The parallel group number.
Field Name: QW0221GN

PLANNED(RUN) DEGREE
The planned degree of parallelism at run time. The value in this field is equal to the value in PLANNED(BIND) DEGREE unless the statement contains host variables.
Field Name: QW0221RD

ACTUAL(RUN) DEGREE
The actual degree of parallelism at run time, taking into account only those DB2 members that have enough buffer pool storage.
Field Name: QW0221AD

REPEATING SECTIONS
The number of repeating sections contained in this record.
Field Name: QW0221N

TYPE OF PARALLELISM
The type of parallelism:
- I/O I/O query parallelism
- SYS Sysplex query parallelism
Field Name: QW0221MO

NUMBER OF MEMBERS
The number of DB2 members on which a query was executed during sysplex query parallel processing.
Field Name: QW0221XC

REASON
The reason for deriving the planned (runtime) degree of parallelism:
- NORMAL The planned runtime degree is derived from the planned bind time degree.
- HOSTVAR Host variable partitioning,
- NO ESA No ESA sort support.
- CURSOR The cursor might be used for an update or deletion.
- EMPTY The parallel group is empty.
- MVS/ESA MVS/ESA enclave services are not available.
Field Name: QW0221RN

RECORD
The position of this record in the series of IFCID 221 records.
Field Name: QW0221TR

OF
IFCID 221 - Parallel Data

The total number of IFCID 221 records in this series.

Field Name: QW0221NR

HI/LO PARTITION TYPE

Type of partition for low and high pages in the partition range. Possible values are:

LOGICAL
Logical low and high pages.

PHYSICAL
Physical low and high pages.

Field Name: QW0221ZZ
IFCID 221 - Section Type D

This topic shows detailed information about “Record Trace - IFCID 221 - Section Type D”.

Record trace - IFCID 221 - Section Type D

The field labels shown in the following sample layout of “Record Trace - IFCID 221 - Section Type D” are described in the following section.

LOW PAGE RANGE
If the partitioning scheme uses a page range, the low page number of the page range.

Field Name: QW0221PL

STATUS
The status of this partition range:

NORMAL A parallel task is created for this partition range.

EMPTY No parallel task is created for this page range.

Field Name: QW0221AN

LOW KEY RANGE
If the partitioning scheme uses a key range, the first 240 bytes of the low boundary key range.

Field Name: QW0221KL

HIGH PAGE RANGE
If the partitioning scheme uses a page range, the high page number of the page range.

Field Name: QW0221PH

HIGH KEY RANGE
If the partitioning scheme uses a key range, the first 240 bytes of the high boundary key range.

Field Name: QW0221KH
IFCID 222 - Parallel Group Elapsed Time

This topic shows detailed information about “Record Trace - IFCID 222 - Parallel Group Elapsed Time”.

This record contains parallel group elapsed time information.

Record trace - IFCID 222 - Parallel Group Elapsed Time

The field labels shown in the following sample layout of “Record Trace - IFCID 222 - Parallel Group Elapsed Time” are described in the following section.

<table>
<thead>
<tr>
<th>RECORD</th>
<th>LOCATION</th>
<th>PACKAGE</th>
<th>PROGRAM</th>
<th>CON.TOKEN</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 OF 1</td>
<td>M05EC003</td>
<td>PRODCOLL</td>
<td>CSF35P04</td>
<td>X'15B995940B8DACAA'</td>
</tr>
<tr>
<td>STATEMENT NO: 84</td>
<td>QUERY BLOCK NO: 1</td>
<td>PARALLEL GROUP NO: 1</td>
<td>REPEAT.GRPS: 2</td>
<td></td>
</tr>
<tr>
<td>PIPE CREATION: 12/18/08 16:35:47.148952</td>
<td>PIPE TERMINATION: 12/18/08 16:35:47.493512</td>
<td>PIPE ELAPSED: 0.344560</td>
<td></td>
<td></td>
</tr>
<tr>
<td>QW0222SM 0 QW0222CS 0 QW0222PR 103 QW022200 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>QW0222SR 21 QW0222OR 0 QW0222CT 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SUB-PIPE CREATION: 12/18/08 16:35:47.242775</td>
<td>SUB-PIPE ELAPSED: 0.076436</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SUB-PIPE TERMINATION: 12/18/08 16:35:47.319211</td>
<td>TASK TOKEN: X'7B450138'</td>
<td>MEMBER: 'BLANK'</td>
<td></td>
<td></td>
</tr>
<tr>
<td>QW0222SR 82 QW0222OR 0 QW0222CT 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>RECORD</th>
<th>LOCATION</th>
<th>PACKAGE</th>
<th>PROGRAM</th>
<th>CON.TOKEN</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 OF 1</td>
<td>M05EC003</td>
<td>PRODCOLL</td>
<td>CSF35P04</td>
<td>X'15B995940B8DACAA'</td>
</tr>
<tr>
<td>STATEMENT NO: 84</td>
<td>QUERY BLOCK NO: 1</td>
<td>PARALLEL GROUP NO: 1</td>
<td>REPEAT.GRPS: 2</td>
<td></td>
</tr>
<tr>
<td>PIPE CREATION: 12/18/08 16:35:47.148952</td>
<td>PIPE TERMINATION: 12/18/08 16:35:47.493512</td>
<td>PIPE ELAPSED: 0.344560</td>
<td></td>
<td></td>
</tr>
<tr>
<td>QW0222SM 0 QW0222CS 0 QW0222PR 103 QW022200 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>QW0222SR 21 QW0222OR 0 QW0222CT 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SUB-PIPE CREATION: 12/18/08 16:35:47.242775</td>
<td>SUB-PIPE ELAPSED: 0.076436</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SUB-PIPE TERMINATION: 12/18/08 16:35:47.319211</td>
<td>TASK TOKEN: X'7B450138'</td>
<td>MEMBER: 'BLANK'</td>
<td></td>
<td></td>
</tr>
<tr>
<td>QW0222SR 82 QW0222OR 0 QW0222CT 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**RECORD**

The position of this record in the series of IFCID 222 records.

**Field Name:** QW0222TR

**OF**

The total number of IFCID 222 records in this series.

**Field Name:** QW0222NR

**LOCATION**

The location name or RDB name.

**Field Name:** QW0222LN

**PACKAGE**

The package collection ID.

**Field Name:** QW0222PC

**PROGRAM**

The program name.

**Field Name:** QW0222PN

**CON.TOKEN**

The timestamp (consistency token).

**Field Name:** QW0222TS

**STATEMENT NO**

The statement number.

**Field Name:** QW0222SN

**QUERY BLOCK NO**

The query block number.

**Field Name:** QW0222QN

40-662 OMEGAMON XE for DB2 PE & PM: Report Reference
PARALLEL GROUP NO
The parallel group number.
Field Name: QW0222GN

REPEAT.GRPS
The number of repeat groups in the section.
Field Name: QW0222RN

PIPE CREATION
The time of pipe creation in DB2 timestamp format.
Field Name: QW0222PS

PIPE TERMINATION
The time of pipe termination in DB2 timestamp format.
The elapsed time between pipe creation and pipe termination in DB2 timestamp format.
Field Name: QW0222PE

SUB-PIPE CREATION
The time of subpipe creation in DB2 timestamp format.
Field Name: QW0222SS

SUB-PIPE ELAPSED
The elapsed time between subpipe creation and subpipe termination in DB2 timestamp format.
The time of subpipe termination in DB2 timestamp format.
Field Name: QW0222SE

TASK TOKEN
The task token associated with the subpipe.
Field Name: QW0222TK

MEMBER
The name of the DB2 member that supplies the data.
Field Name: QW0222SM
IFCID 223 - Lock Avoidance Detail

This topic shows detailed information about “Record Trace - IFCID 223 - Lock Avoidance Detail”.

This record shows lock avoidance information for each successful test.

Record trace - IFCID 223 - Lock Avoidance Detail

The field labels shown in the following sample layout of “Record Trace - IFCID 223 - Lock Avoidance Detail” are described in the following section.

LOCK RES TYPE

The resource type being accessed.

Field Name: QW0223KT

DBID

The database ID.

Field Name: QW0223KD

OBID

Object id of the page set or table record.

Field Name: QW0223KP

TABLE_SPACE_TYPE

The type of the table space:

L Non-EA large table
N Non-large table
V EA-enabled large table

Field Name: QW0223TY

RESOURCE ID

The ID of the small resource.

Field Name: QW0223KR
IFCID 224 - Select Procedure Bypassed

This topic shows detailed information about “Record Trace - IFCID 224 - Select Procedure Bypassed”.

This record is written at the end of a unit of work. It records the total columns for which an invalid select procedure was encountered. Invalid select procedures are bypassed by DB2 and can cause performance degradation.

Record trace - IFCID 224 - Select Procedure Bypassed

The field labels shown in the following sample layout of “Record Trace - IFCID 224 - Select Procedure Bypassed” are described in the following section.

SPROC BYPASSED 'BLANK'
NETWORKID: DEIBMIPS LUNAME: IPSAR721 LUWSEQ: 1
PACKAGE : PLANNAM2
COLLECTION: COLLECTIONNAME0002
COLUMNS : 128

COLUMNS
The select procedure bypass column count. This is the total number of columns (rows * columns) for which a select procedure was bypassed because the select procedure was invalidated by applying service to DB2.

Field Name: QW0224CL

PACKAGE
Package name.

Field Name: QW0224PN

COLLECTION ID
Collection identifier.

Field Name: QW0224CI
IFCID 225 - System Storage Usage

This topic shows the data available for IFCID 225.

- **IFCID 225 - Address Space Summary - DBM1** on page 40-667
  This topic shows detailed information about “Record Trace - IFCID 225 - Address Space Summary - DBM1”.

- **IFCID 225 - Address Space Summary - DIST** on page 40-668
  This topic shows detailed information about “Record Trace - IFCID 225 - Address Space Summary - DIST”.

- **IFCID 225 - Statement Cache / XPROC Detail** on page 40-672
  This topic shows detailed information about “Record Trace - IFCID 225 - Statement Cache / XPROC Detail”.

- **IFCID 225 - Shared/Common Storage Summary** on page 40-675
  This topic shows detailed information about “Record Trace - IFCID 225 - Shared/Common Storage Summary”.

- **IFCID 225 - Storage Pool Details** on page 40-679
  This topic shows detailed information about “Record Trace - IFCID 225 - Storage Pool Details”.

- **IFCID 225 - Thread Information** on page 40-681
  This topic shows detailed information about “Record Trace - IFCID 225 - Thread Information”.
IFCID 225 - Address Space Summary - DBM1

This topic shows detailed information about “Record Trace - IFCID 225 - Address Space Summary - DBM1”.

Note: This report has the same layout as “IFCID 225 - Address Space Summary - DIST” on page 40-668.
IFCID 225 - Address Space Summary - DIST

This topic shows detailed information about “Record Trace - IFCID 225 - Address Space Summary - DIST”.

Record trace - IFCID 225 - Address Space Summary - DIST

The field labels shown in the following sample layout of “Record Trace - IFCID 225 - Address Space Summary - DIST” are described in the following section.

**ADDRESS SPACE SUMMARY - DIST**

**EXTENDED REGION SIZE (MAX)**

The maximum amount of MVS private storage available above the 16 MB line.

Field Name: QW0225RG

**24-BIT LOW PRIVATE**

The amount of private MVS storage below the 16 MB line. This storage is obtained from bottom upward, usually for unauthorized programs.

Field Name: QW0225LO

**24-BIT HIGH PRIVATE**

The amount of private MVS storage below the 16 MB line. This storage is obtained from top downward, usually for authorized programs.

Field Name: QW0225HI

**31-BIT EXTENDED LOW PRIVATE**

The amount of private MVS storage above the 16 MB line. This storage is obtained from bottom upward, usually for unauthorized programs.

Field Name: QW0225EL

**31-BIT EXTENDED HIGH PRIVATE**

The amount of private MVS storage above the 16 MB line. This storage is obtained from top downward, usually for authorized programs.

Field Name: QW0225EH

**CURR HIGH ADDR 24-BIT PRIV REGION**

The current high address of the 24-bit private region.

Field Name: QW0225TP
CURR HIGH ADDR 31-BIT PRIV REGION
The current high address of the 31-bit private region.
Field Name: QW0225EP

31-BIT RESERVED FOR MUST COMPLETE
Storage reserved for operation that must complete before DB2 is allowed to stop.
Field Name: QW0225CR

31-BIT RESERVED FOR MVS
The amount of storage available for operating system activity.
Field Name: QW0225MV

STORAGE CUSHION WARNING TO CONTRACT
The amount of free storage, in megabytes, available in the DBM1 data space.
Field Name: QW0225SO

TOTAL 31-BIT GETMAINED STACK
Total GETMAINED storage allocated for program stack use.
Field Name: QW0225GS

TOTAL 31-BIT STACK IN USE
The amount of stack storage that is in use.
Field Name: QW0225SU

TOTAL 31-BIT VARIABLE POOL
Total storage used by all variable pools. This includes storage used by:
- System agents
- Local agents
- RID pool
- Pipe manager subpool
- Local dynamic statement cache control blocks
- Local dynamic statement cache statement pool
- Buffer and data manager trace tables
- A list of objects in restricted state including the new PRO state. If consumption of this storage pool is high, review restrictive exception state of database objects and check whether they can be resolved or reduced.
Field Name: QW0225VR

TOTAL 31-BIT FIXED POOL
Total amount of fixed storage.
Field Name: QW0225FX

TOTAL 31-BIT GETMAINED
Total storage acquired by GETMAIN. This includes space for virtual pools, EDM pool, compression dictionary, castout buffers, and the data space looksaside buffer, hiperpool control blocks, and data space buffer pool control blocks.

This figure can be different from the sum of GETMAIN storage items shown in the statistics DBM1 storage, because DB2 does not produce grouping statistics for all GETMAIN storage.

Field Name: QW0225GM

AMOUNT OF AVAILABLE 31-BIT

The total amount of storage available for storage manager pools.

Field Name: QW0225AV

SYSTEM AGENT STACK STORAGE IN USE

The amount of 31-bit stack storage that is in use for system agents. This is a subset of QW0225SU.

Field Name: QW0225SS

TOTAL 64-BIT VARIABLE POOL

Amount of variable storage available above the 2 GB bar.

Field Name: QW0225VA

TOTAL 64-BIT FIXED

The total amount of fixed storage above the 2 GB bar.

Field Name: QW0225FA

TOTAL 64-BIT GETMAINED

Total storage acquired by GETMAIN. This includes space for the compression dictionary, and statement and DBD cache that can be used by the Environmental Descriptor Manager (EDM).

This figure can be different from the sum of GETMAIN storage items shown in the statistics DBM1 storage, because DB2 does not produce grouping statistics for all GETMAIN storage.

Field Name: QW0225GA

TOTAL 64-BIT PRIVATE FOR STOR MANAG

Total 64-bit storage allocated for storage manager control structures.

Field Name: QW0225SM

REAL 4K FRAMES IN USE

Number of real-storage frames (4K) in use for 31- and 64-bit private pools.

Field Name: QW0225RL

AUXILIARY SLOTS IN USE

Number of auxiliary slots (4K) in use for 31- and 64-bit private pools.

Field Name: QW0225AX

64-BIT REAL 4K FRAMES IN USE

The number of real 4K frames in use for 64-bit private pools.
Note: This value is available from z/OS V1.11.

Field Name: QW0225HVPGESINREAL

64-BIT 4K AUX SLOTS IN USE
The number of auxiliary 4K slots in use for 64-bit private pools.

Note: This value is available from z/OS V1.11.
Field Name: QW0225HVAUXSLOTS

ABOVE VALUE W/O BP STORAGE
Number of real-storage frames (4K) in use for 64-bit private pools. This is a subset of QW0225HVPagesInReal and does not include buffer pool storage.

Note: This field is available in z/OS 1.10 (and maintenance) or later.
Field Name: QW0225PRISTG_REAL

ABOVE VALUE W/O BP STORAGE
Number of auxiliary slots (4K) in use for 64-bit private pools. This does not include buffer pool storage. This field only includes auxiliary slots occupied by pages that are paged out.

Note: This field is available in z/OS 1.10 (and maintenance) or later.
Field Name: QW0225PRISTG_AUX

HWM 64-BIT REAL 4K FRAMES IN USE
The number of real 4K frames in use for 64-bit private pools.

Note: This value is available from z/OS V1.11.
Field Name: QW0225HVGPAGESINREAL

HWM 64-BIT AUX SLOTS IN USE
High water mark for the number of auxiliary 4K slots in use for 64-bit private pools.

Note: This value is available from z/OS V1.11.
Field Name: QW0225HVGAUXSLOTS

QW0225CTLP (S)
This field is for IBM service.
Field Name: QW0225CTLP

QW0225CTLS (S)
This field is for IBM service.
Field Name: QW0225CTLS
IFCID 225 - Statement Cache / XPROC Detail

IFCID 225 - Statement Cache / XPROC Detail

This topic shows detailed information about “Record Trace - IFCID 225 - Statement Cache / XPROC Detail”.

Record trace - IFCID 225 - Statement Cache / XPROC Detail

The field labels shown in the following sample layout of “Record Trace - IFCID 225 - Statement Cache / XPROC Detail” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALLOCATED STOR FOR DYN SQL STMTS</td>
<td>The total shareable storage allocated for dynamic SQL statements used by active threads.</td>
</tr>
<tr>
<td>REQUESTED STOR FOR DYN SQL STMTS</td>
<td>The total shareable storage requested for dynamic SQL statements used by active threads.</td>
</tr>
<tr>
<td>ALLOCATED STOR FOR STATIC SQL STMTS</td>
<td>The total shareable storage allocated for static SQL statements.</td>
</tr>
<tr>
<td>HWM REQUESTED STOR FOR DYN SQL STMTS</td>
<td>A statistics interval high-water mark of requested shareable storage for dynamic SQL statements used by active threads.</td>
</tr>
<tr>
<td>TOTAL 31-BIT XPROC DYNAMIC SQL</td>
<td>The amount of storage allocated for the local cache storage pool below the bar.</td>
</tr>
<tr>
<td>ALLOCATED 31-BIT XPROC DYNAMIC SQL</td>
<td>The amount of storage used for thread copies in the local cache storage pool below the bar. This is a subset of the total allocated storage for thread copies.</td>
</tr>
</tbody>
</table>

Note: For DB2 10, the storage is allocated for executable code sequences of dynamic SQL statements.
**Field Name:** QW0225LS

**TOTAL 31-BIT XPROC STATIC SQL**

The amount of storage allocated below the bar for executable code sequences of static SQL statements.

**Field Name:** QW0225SX

**HWM ALLOCATED 31-BIT XPROC DYNAMIC SQL**

A statistics interval high-water mark of allocated storage for thread copies in the local cache storage pool below the bar.

**Note:** For DB2 10, the high water mark is related to executable code sequences of dynamic SQL statements.

**Field Name:** QW0225HS

**STATEMENTS IN 64-BIT AGENT LOCAL POOLS (ALP)**

The number of dynamic SQL local cache statements used by active threads.

- For DB2 10 or later, this value is related to shared agent local variable pools above the bar.
- Prior to DB2 10, this value is related to the local cache storage pool below the bar.

**Field Name:** QW0225LC

**HWM STMT COUNT IN 64-BIT ALP AT HIGH STOR TIME**

The number of dynamic SQL local cache statements used by active threads at high storage time.

- For DB2 10 or later, this value is related to shared agent local variable pools above the bar.
- Prior to DB2 10, this value is related to the local cache storage pool below the bar.

**Field Name:** QW0225HC

**ALLOCATED STMT CACHE IN 64-BIT ALP**

The total non-shareable storage requested for dynamic SQL statements used by active threads.

**Note:** For DB2 10 or later, this value is related to shared agent local variable pools above the bar.

**Field Name:** QW0225L2

**HWM ALLOCATED STMT CACHE 64-BIT ALP**

A statistics interval high-water mark of requested non-shareable storage for dynamic SQL statements used by active threads.

**Note:** For DB2 10 or later, this value is related to shared agent local variable pools above the bar.

**Field Name:** QW0225H2

**TIMESTAMP OF HWM AFTER LAST 225 REC**

The timestamp at high-water storage.

**Field Name:** QW0225HT
IFCID 225 - Statement Cache / XPROC Detail

TOTAL 64-BIT STMT CACHE BLKS 2G
  The total statement cache storage blocks above the bar (64-bit shared variable pool).
  Field Name: QW0225S2

QW0225F1
  This field is for IBM service.
  Field Name: QW0225F1

QW0225F2
  This field is for IBM service.
  Field Name: QW0225F2
IFCID 225 - Shared/Common Storage Summary

Record trace - IFCID 225 - Shared/Common Storage Summary

This topic shows detailed information about “Record Trace - IFCID 225 - Shared/Common Storage Summary”.

The field labels shown in the following sample layout of “Record Trace - IFCID 225 - Shared/Common Storage Summary” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>31-BIT COMMON FIXED POOL STORAGE</td>
<td>1060864</td>
</tr>
<tr>
<td>31-BIT COMMON GETMAINED STORAGE</td>
<td>96501</td>
</tr>
<tr>
<td>31-BIT COMMON VARIABLE POOL STORAGE</td>
<td>2629632</td>
</tr>
<tr>
<td>64-BIT COMMON FIXED POOL STORAGE</td>
<td>200512</td>
</tr>
<tr>
<td>64-BIT COMMON GETMAINED STORAGE</td>
<td>803366368</td>
</tr>
<tr>
<td>64-BIT COMMON VARIABLE POOL STORAGE</td>
<td>29487104</td>
</tr>
<tr>
<td>64-BIT COMMON STORAGE-STOR MGR CTRL</td>
<td>268435456</td>
</tr>
<tr>
<td>64-BIT SHARED VARIABLE POOL STORAGE</td>
<td>268435456</td>
</tr>
<tr>
<td>64-BIT SHARED FIXED POOL STORAGE</td>
<td>268435456</td>
</tr>
<tr>
<td>64-BIT SHARED GETMAINED STORAGE</td>
<td>268435456</td>
</tr>
<tr>
<td>64-BIT SHARED SYSTEM AS IN USE</td>
<td>75759616</td>
</tr>
<tr>
<td>64-BIT SHARED NON-SYSTEM AS</td>
<td>805306368</td>
</tr>
<tr>
<td>64-BIT SHARED STORAGE-STOR MGR CTRL</td>
<td>268435456</td>
</tr>
<tr>
<td>64-BIT SHARED SYSTEM AGENT STACK (AS)</td>
<td>268435456</td>
</tr>
<tr>
<td>64-BIT SHARED NON-SYSTEM AS IN USE</td>
<td>75759616</td>
</tr>
<tr>
<td>SHARED MEMORY OBJECTS</td>
<td>12</td>
</tr>
<tr>
<td>64-BIT SHARED MEMORY PAGES</td>
<td>754974720</td>
</tr>
<tr>
<td>64-BIT PAGES PAGED IN FROM AUX STOR</td>
<td>167232</td>
</tr>
<tr>
<td>64-BIT SHARED STG REAL 4K FRMS IN USE:</td>
<td>803</td>
</tr>
<tr>
<td>64-BIT STACK STG REAL 4K FRMS IN USE:</td>
<td>200</td>
</tr>
<tr>
<td>64-BIT COMMON STG REAL 4K FRMS IN USE:</td>
<td>142</td>
</tr>
<tr>
<td>LOG MGR WRITE BUFFER FRAMES IN REAL:</td>
<td>1010</td>
</tr>
<tr>
<td>LOG MGR WRITE BUFFER FRAMES IN AUX:</td>
<td>1</td>
</tr>
<tr>
<td>QW0225_FC</td>
<td>1</td>
</tr>
<tr>
<td>QW0225_REALAVAIL</td>
<td>1</td>
</tr>
<tr>
<td>QW0225_REALAVAILLO</td>
<td>490</td>
</tr>
<tr>
<td>QW0225_ESQAS</td>
<td>146489344</td>
</tr>
<tr>
<td>QW0225_ECSA_HWM</td>
<td>142820884</td>
</tr>
<tr>
<td>QW0225CTGP</td>
<td>OFF</td>
</tr>
</tbody>
</table>

31-BIT COMMON FIXED POOL STORAGE

The amount of storage allocated for 31-bit common fixed pool storage.

Field Name: QW0225FC

31-BIT COMMON VARIABLE POOL STORAGE

The amount of storage allocated for 31-bit common variable pool storage.

Field Name: QW0225VC

31-BIT COMMON GETMAINED STORAGE

The amount of storage allocated for 31-bit common getmained storage.

Field Name: QW0225GC

EXTENDED CSA SIZE

The size of the common storage area (CSA) above the 16 MB line.

Field Name: QW0225EC

64-BIT COMMON FIXED POOL STORAGE

The amount of storage allocated for 64-bit common fixed pool storage.

Field Name: QW0225FCG

64-BIT COMMON VARIABLE POOL STORAGE

The amount of storage allocated for 64-bit common variable pool storage.

Field Name: QW0225VCG
IFCID 225 - Shared/Common Storage Summary

64-BIT COMMON GETMAINED STORAGE
The amount of storage allocated for 64-bit common getmained storage.
Field Name: QW0225GCG

64-BIT COMMON STORAGE-STOR MGR CTRL
The amount of storage allocated for 64-bit common storage for storage manager control structures.
Field Name: QW0225SMC

64-BIT SHARED VARIABLE POOL STORAGE
The amount of virtual shared variable storage above the 2 GB bar.
Field Name: QW0225SV

64-BIT SHARED FIXED POOL STORAGE
The amount of total fixed virtual shared storage above the 2 GB bar.
Field Name: QW0225SF

64-BIT SHARED GETMAINED STORAGE
The amount of virtual shared storage acquired by GETMAIN above the 2 GB bar.
Field Name: QW0225SG

64-BIT SHARED STORAGE-STOR MGR CTRL
The amount of 64-bit shared storage allocated for storage manager control structures.
Field Name: QW0225SMS

64-BIT SHARED SYSTEM AGENT STACK (AS)
The amount of 64-bit shared storage allocated for system agent stack use.
Field Name: QW0225GSG_SYS

64-BIT SHARED SYSTEM AS IN USE
The amount of 64-bit shared system agent stack that is in use.
Field Name: QW0225SUG_SYS

64-BIT SHARED NON-SYSTEM AS
The amount of 64-bit shared storage allocated for non-system agent stack use.
Field Name: QW0225GSG

64-BIT SHARED NON-SYSTEM AS IN USE
The amount of 64-bit shared non-system agent stack that is in use.
Field Name: QW0225SUG

SHARED MEMORY OBJECTS
The number of shared memory objects allocated for this MVS LPAR.
Field Name: QW0225SHRNOMB

64-BIT SHARED MEMORY PAGES
IFCID 225 - Shared/Common Storage Summary

The number of 64-bit shared memory pages allocated for this MVS LPAR (this count includes hidden pages).

**Field Name:** QW0225SHRPAGES

**HWM FOR 64-BIT SHARED BYTES**

High water mark for number of 64-bit shared bytes for this MVS LPAR.

**Field Name:** QW0225SHRGBYTES

**64-BIT SHARED PAGES BACKED IN REAL**

The number of 64-bit shared pages backed in real storage (4K pages) for this MVS LPAR.

**Field Name:** QW0225SHRINREAL

**AUX SLOTS USED FOR 64-BIT SHARED STOR**

The number of auxiliary slots used for 64-bit shared storage for this MVS LPAR.

**Field Name:** QW0225SHRAUXSLOTS

**64-BIT PAGES PAGED IN FROM AUX STOR**

The number of 64-bit shared pages paged in from auxiliary storage for this MVS LPAR.

**Field Name:** QW0225SHRPAGEINS

**64-BIT PAGES PAGED OUT TO AUX STOR**

The number of 64-bit shared pages paged out to auxiliary storage for this MVS LPAR.

**Field Name:** QW0225SHRPAGEOUTS

**64-BIT SHARED STG REAL 4K FRMS IN USE**

The number of real-storage frames (4K) in use for 64-bit shared storage. This does not include shared stack storage. This is recorded at the subsystem level.

**Note:** This field is available in z/OS 1.10 (and maintenance) or later.

**Field Name:** QW0225SHRSTG_REAL

**64-BIT SHARED STG 4K AUX SLOTS IN USE**

The number of auxiliary slots (4K) in use for 64-bit shared storage. This does not include shared stack storage. This is recorded at the subsystem level. This field only includes auxiliary slots occupied by pages that are paged out.

**Note:** This field is available in z/OS 1.10 (and maintenance) or later.

**Field Name:** QW0225SHRSTG_AUX

**64-BIT STACK STG REAL 4K FRMS IN USE**

The number of real-storage frames (4K) in use for 64-bit shared stack storage. This is recorded at the subsystem level.

**Note:** This field is available in z/OS 1.10 (and maintenance) or later.

**Field Name:** QW0225SHRSTKSTG_REAL
IFCID 225 - Shared/Common Storage Summary

64-BIT STACK STG 4K AUX SLOTS IN USE
The number of auxiliary slots (4K) in use for 64-bit shared stack storage. This is recorded at the subsystem level. This field only includes auxiliary slots occupied by pages that are paged out.

Note: This field is available in z/OS 1.10 (and maintenance) or later.
Field Name: QW0225SHRSTKSTG_AUX

64-BIT COMMON STG REAL 4K FRMS IN USE
The number of real-storage frames (4K) in use for 64-bit common storage. This is recorded at the subsystem level.

Note: This field is available in z/OS 1.10 (and maintenance) or later.
Field Name: QW0225COMSTG_REAL

64-BIT COMMON STG 4K AUX SLOTS IN USE
The number of auxiliary slots (4K) in use for 64-bit common storage. This is recorded at the subsystem level. This field only includes auxiliary slots occupied by pages that are paged out.

Note: This field is available in z/OS 1.10 (and maintenance) or later.
Field Name: QW0225COMSTG_AUX

LOG MGR WRITE BUFFER FRAMES IN REAL
The number of real-storage frames (4K) in the 64-bit common area in use for Log Manager write buffers.
Field Name: QW0225_LMWRITE_REAL

LOG MANAGER CONTROL FRAMES IN REAL
The number of real-storage frames (4K) in the 64-bit common area in use for Log Manager control blocks.
Field Name: QW0225_LMCTRL_REAL

LOG MGR WRITE BUFFER FRAMES IN AUX
The number of auxiliary slots (4K) in the 64-bit common area in use for Log Manager write buffers.
Field Name: QW0225_LMWRITE_AUX

LOG MANAGER CONTROL FRAMES IN AUX
The Number of auxiliary slots (4K) in the 64-bit common area in use for Log Manager control blocks.
Field Name: QW0225_LMCTRL_AUX

QW0225_ECSA_CONV
This field is for IBM service.
Field Name: QW0225_ECSA_CONV

QW0225LFAREA
This field is for IBM service.
Field Name: QW0225LFAREA
IFCID 225 - Storage Pool Details

This topic shows detailed information about “Record Trace - IFCID 225 - Storage Pool Details”.

Record trace - IFCID 225 - Storage Pool Details

The field labels shown in the following sample layout of “Record Trace - IFCID 225 - Storage Pool Details” are described in the following section.

31-BIT DBM1 PRIVATE VARIABLE POOLS:

- AGENT LOCAL STORAGE: 2162688
- SYSTEM AGENT STORAGE: 2109440
- BUFFER MANAGER STORAGE BLOCKS: 651264

64-BIT POOLS:

- SHARED AGENT LOCAL (VARIABLE POOL): 27615232
- SHARED SYSTEM AGENT (VARIABLE POOL): 26210304
- RID POOL STORAGE (FIXED POOL): 0
- COMPRESSION DICT (DBM1 PRIVATE): 0
- ARRAY VARIABLE STORAGE: 0

AGENT LOCAL STORAGE

The amount of storage allocated for agent-related local storage. This storage is used for operations such as sort.

Background and Tuning Information

Sorting requires a large amount of virtual storage because there can be multiple copies of the data being sorted at a given time.

DB2 Sort uses two kinds of storage pool for various internal control structures and data records, an agent-related local storage pool and a global sort pool. To take advantage of the 64-bit addressability for larger storage pool, some high level sort control structures are kept in agent-related storage below the 2 GB bar, which contain 64-bit pointers to areas in the global sort pool above the 2 GB bar. The sort pool above 2 GB contains sort tree nodes and data buffers.

Field Name: QW0225AL

SYSTEM AGENT STORAGE

Storage used by system agents.

Field Name: QW0225AS

BUFFER MANAGER STORAGE BLOCKS

Storage used for page set control blocks.

Field Name: QW0225BB

SHARED AGENT LOCAL (VARIABLE POOL)

The amount of storage allocated for agent-related 64-bit local storage (DB2 field: QW0225ALG).

Field Name: QW0225ALG

SHARED SYSTEM AGENT (VARIABLE POOL)

The amount of 64-bit storage used by system agents (DB2 field: QW0225ASG).

Field Name: QW0225ASG

RID POOL STORAGE (FIXED POOL)

Storage for RID list processing such as list prefetch, index ANDing, and ORing.
IFCID 225 - Storage Pool Details

Note:
- Prior to DB2 10, this value is shown in the “DBM1 AND MVS BELOW 2 GB” block.
- For DB2 10 or later, this value is shown in the “DBM1 STORAGE ABOVE 2 GB” block.

Field Name: QW0225RP

COMPRESSION DICT (DBM1 PRIVATE)
Storage space allocated for the compression dictionary.

Field Name: QW0225CD

ARRAY VARIABLE STORAGE
The amount of storage in use for array variables.

Field Name: QW0225AR
IFCID 225 - Thread Information

This topic shows detailed information about “Record Trace - IFCID 225 - Thread Information”.

Record trace - IFCID 225 - Thread Information

The field labels shown in the following sample layout of “Record Trace - IFCID 225 - Thread Information” are described in the following section.

<table>
<thead>
<tr>
<th>THREAD INFORMATION</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ACTIVE THREADS</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>CASTOUT ENGINES</td>
<td>0</td>
<td>16</td>
</tr>
<tr>
<td>GBP WRITE ENGINES</td>
<td>0</td>
<td>23</td>
</tr>
<tr>
<td>P-LOCK/NOTIFY EXIT ENGINES</td>
<td>0</td>
<td>22</td>
</tr>
</tbody>
</table>

ACTIVE THREADS

The number of active allied threads.

Field Name: QW0225AT

ACTIVE AND DISCONNECTED DBATS

The number of active and disconnected DBAT threads.

Field Name: QW0225DB

CASTOUT ENGINES

Number of engines available for data-sharing castout processing.

Field Name: QW0225CE

DEFERRED WRITE ENGINES

Number of engines used for deferred write operations.

Field Name: QW0225DW

GBP WRITE ENGINES

Number of engines for group buffer pool writes.

Field Name: QW0225GW

PREFETCH ENGINES

Number of engines used for sequential, list, and dynamic prefetch.

Field Name: QW0225PF

P-LOCK/NOTIFY EXIT ENGINES

Number of data sharing P-Lock engines and Notify Exit engines.

Field Name: QW0225PL

PARALLEL CHILDTHREADS

The number of active parallel child threads.

Field Name: QW0225PT
This topic shows detailed information about “Record Trace - IFCID 226 - Page Latch Contention Start”.

This IFCID records the beginning of an agent suspend to wait for a page latch that is currently held by another agent.

**Record trace - IFCID 226 - Page Latch Contention Start**

The field labels shown in the following sample layout of “Record Trace - IFCID 226 - Page Latch Contention Start” are described in the following section.

```plaintext
PAGE LATCH --> 'BLANK'
CONTENTION NETWORKID: DEIBMIPS LUNAME: IPSAQ811 LUWSEQ: 1
DBID: 1 PSID: 2
PAGE NUMBER: X'000003' LATCH TYPE: S
TABLE_SPACE_TYPE: N
BUFFERPOOL ID: 4	ACE TOKEN: 2
```

**DBID**

The database ID. Deduced from the DB2 fields QW0226DB, QW0105DN or QW0107DN.

When present, the database name is shown, otherwise the decimal identifier from QW0226DB is shown or N/A when this value is 0.

Field Name: RT0226DB

**PSID**

The page set object identifier. When present, this is the page set object name, otherwise the decimal identifier from QW0226OB is shown.

Field Name: RT0226OB

**PAGE NUMBER**

The number of the page being read or written. If the value in TABLE_SPACE_TYPE is L or V, the page number covers 4 bytes instead of 3.

Field Name: QW0226PN

**LATCH TYPE**

The type of latch.

Field Name: QW0226F

**BUFFERPOOL ID**

The buffer pool internal identifier. The values 0 through 49 are the identifiers for BP0 through BP49. The values 80 through 89 are the identifiers for BP32K through BP32K9.

Field Name: QW0226BP

**ACE TOKEN**

The agent control element token of the requester.

Field Name: QW0226AC

**TABLE SPACE TYPE**

The type of the table space:
### IFCID 226 - Page Latch Contention Start

| L | Non-EA large table |
| N | Non-large table    |
| V | EA-enabled large table |

**Field Name:** QW0226FG
This topic shows detailed information about “Record Trace - IFCID 227 - Page Latch Contention End”.

This IFCID records the end of an agent suspend wait for a page latch that was currently held by another agent.

**Record trace - IFCID 227 - Page Latch Contention End**

The field labels shown in the following sample layout of “Record Trace - IFCID 227 - Page Latch Contention End” are described in the following section.

```
PAGE LATCH -- 'BLANK'
CONTENTION NETWORKID: DEIBMIPS LUNAME: IPSAQ811 LUWSEQ: 1
DBID: 1 PSID: 2
PAGE NUMBER: X'000003' CANCEL STATUS: Y
ACE TOKEN: 2 TABLE SPACE_TYPE: N
```

**DBID**

The database ID. Deduced from the DB2 fields QW0227DB, QW0105DN or QW0107DN.

When present, the database name is shown, otherwise the decimal identifier from QW0227DB is shown or N/A when this value is 0.

*Field Name:* RT0227DB

**PSID**

The page set object identifier. When present this is the page set object name, otherwise it is the decimal identifier from QW0227OB.

*Field Name:* RT0227OB

**PAGE NUMBER**

The number of the page being read or written. If the value in TABLE_SPACE_TYPE is L or V, the page number covers 4 bytes instead of 3.

*Field Name:* QW0227PN

**CANCEL STATUS**

Indicates whether the latch requester was canceled.

*Field Name:* QW0227F

**ACE TOKEN**

The agent control element token of the requester.

*Field Name:* QW0227AC

**TABLE SPACE TYPE**

The type of the table space:

- L Non-EA large table
- N Non-large table
- V EA-enabled large table

*Field Name:* QW0227FG
This topic shows detailed information about “Record Trace - IFCID 228 - Archive Deallocation Start”.

Record trace - IFCID 228 - Archive Deallocation Start

The field labels shown in the following sample layout of “Record Trace - IFCID 228 - Archive Deallocation Start” are described in the following section.

QW0228DV 1 QW0228DI X'C6D9C5C4F1404040'
IFCID 229 - Archive Deallocation End

This topic shows detailed information about “Record Trace - IFCID 229 - Archive Deallocation End”.

Record trace - IFCID 229 - Archive Deallocation End

The field labels shown in the following sample layout of “Record Trace - IFCID 229 - Archive Deallocation End” are described in the following section.

QW0229DV 1 QW0229CC 'BLANK'
IFCID 230 - Group Buffer Pool Attributes

This topic shows detailed information about “Record Trace - IFCID 230 - Group Buffer Pool Attributes”.

Each repeating section contains information about each group buffer pool to which this DB2 data sharing member is currently connected.

Record trace - IFCID 230 - Group Buffer Pool Attributes

The field labels shown in the following sample layout of “Record Trace - IFCID 230 - Group Buffer Pool Attributes” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GROUP BUFFERPOOL ID</td>
<td>Group buffer pool name.</td>
</tr>
<tr>
<td>ERROR FLAGS</td>
<td>Indicates whether errors occurred during data collection:</td>
</tr>
<tr>
<td>ALLOCATED GBPOOL SIZE (4K)</td>
<td>The allocated size of the group buffer pool in 4 KB blocks.</td>
</tr>
<tr>
<td>CURRENT DIR TO DATA RATIO</td>
<td>The current directory entry to data entry ratio.</td>
</tr>
<tr>
<td>CLASS CASTOUT THRESH (%)</td>
<td>The threshold at which the class castout is to be initiated. It is expressed as a percentage of the size of the group buffer pool.</td>
</tr>
<tr>
<td>ACTUAL # OF DIR ENTRIES</td>
<td>The actual number of allocated directory entries.</td>
</tr>
</tbody>
</table>

GROUP BUFFERPOOL ID

GROUP BUFFERPOOL ID

Field Name: QBGBGN

ERROR FLAGS

Field Name: QBGBFLGS

Indicates whether errors occurred during data collection:

- X'00' There were no errors.
- X'01' An error occurred when the group buffer pool attributes were read from the SCA.
- X'02' An addressing error occurred when the DB2 control blocks were accessed. The data collection process did not obtain serialization.

Field Name: QBGBFLGS

ALLOCATED GBPOOL SIZE (4K)

Field Name: QBGBGSZ

The allocated size of the group buffer pool in 4 KB blocks.

CURRENT DIR TO DATA RATIO

Field Name: QBGBGR1

The current directory entry to data entry ratio.

For ALTER GROUPBUFFERPOOL commands, this field reports the value specified in the RATIO keyword.

CLASS CASTOUT THRESH (%)

Field Name: QBGBGCT

The threshold at which the class castout is to be initiated. It is expressed as a percentage of the size of the group buffer pool.

For ALTER GROUPBUFFERPOOL commands, it reports the value specified in the CLASST keyword.

ACTUAL # OF DIR ENTRIES

Field Name: QBGBGCT

The actual number of allocated directory entries.
IFCID 230 - Group Buffer Pool Attributes

Field Name: QBGBGDR

PENDING DIR TO DATA RATIO
The pending directory entry to data entry ratio.

Field Name: QBGBGR2

CLASS CASTOUT THRESH (PGS)
The threshold at which the castout is to be initiated for the group buffer pool. It is expressed as a percentage of the size of the group buffer pool.
For ALTER GROUPBUFFERPOOL commands, it reports the value specified in the GBPOOLT keyword.

Field Name: QBGBGGR

ACTUAL # OF DATA ENTRIES
The actual number of allocated data entries.

Field Name: QBGBDDT

GBP CHECKPOINT INTERVAL (MIN)
The time interval, in minutes, between successive group buffer pool checkpoints.
For ALTER GROUPBUFFERPOOL commands, it reports the value specified in the GBPCHKPT keyword.

Field Name: QBGBGCK

GBP CASTOUT THRESH (%)
The threshold at which the castout is to be initiated for the group buffer pool. It is expressed as a percentage of the size of the group buffer pool.
For ALTER GROUPBUFFERPOOL commands, it reports the value specified in the GBPOOLT keyword.

Field Name: QBGBGGR

DIRECTORY-ENTRY-RECLAIM
The number of times that a page name assignment required that a coupling facility directory entry be reclaimed (stolen).

Field Name: QBGBDRR

DATA-ENTRY-RECLAIM
The number of times that a page name assignment required that a coupling facility data entry be reclaimed (stolen).

Field Name: QBGBDTR

AUTOREC
Indicates whether automatic recovery takes place in the event of a structure failure or a loss of connectivity. When automatic recovery is active, all members of the group are recovered to the group buffer pool.

Field Name: QBGBGAS

TOTAL-CHANGED
The number of allocated data entries that are currently in changed state. This is a snapshot value and is not cumulative.
Field Name: QBGBTCC
XI-DIRECTORY-ENTRY-RECLAIM

The number of times that a directory entry was stolen and one or more XI signals had to be sent because the page in the directory was cached in one or more DB2 buffer pools.

Field Name: QBGBRXI
GBP CACHE

GBP cache attribute. Possible values are:

YES  GBP is used for both data caching and cross-invalidation.
NO   GBP is used for cross-invalidation only.

Field Name: QBGBGCS
MODE

Simplex or duplex mode indicator.

Field Name: QBGBDUP
SEC-GBP ALLOC

The allocated size of the secondary GBP when the GBP is DUPLEX.

This field is not shown when MODE is SIMPLEX.

Field Name: QBGBGSZ2
SEC-GBP ALLOC DIRECTORIES

Number of allocated directory entries in the secondary GBP when MODE is DUPLEX.

Field Name: QBGBGDR2
SEC-GBP DATA ENTRIES

The allocated data entries in the secondary GBP when MODE is DUPLEX.

Field Name: QBGBGDT2
IFCID 231 - Parallel Group Task Time

This topic shows detailed information about “Record Trace - IFCID 231 - Parallel Group Task Time”.

Record trace - IFCID 231 - Parallel Group Task Time

The field labels shown in the following sample layout of “Record Trace - IFCID 231 - Parallel Group Task Time” are described in the following section.

<table>
<thead>
<tr>
<th>Statement NO: 1224</th>
<th>Query block No: 1</th>
<th>Parallel group No: 1</th>
<th>Repeat.GRPS: 2</th>
<th>Record 1 of 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Statement no:</td>
<td>Query block no:</td>
<td>Parallel group no:</td>
<td>Repeat groups:</td>
<td>Record 1 of 1</td>
</tr>
<tr>
<td>STATEMENT NO:</td>
<td>QUERY BLOCK NO:</td>
<td>PARALLEL GROUP NO:</td>
<td>REPEAT.GRPS:</td>
<td>RECORD 1 OF 1</td>
</tr>
<tr>
<td>1224</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>QW0231NG</td>
<td>QW0231NT</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>..........................................................</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TASK SEQ. NUMBER: 1</td>
<td>TASK TOKEN: X'6B9C08D0'</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TASK CREATION: 09/24/08 12:40:57.763105</td>
<td>TASK ELAPSED: 1:08.763455</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TASK TERMINATION: 09/24/08 12:42:06.526560</td>
<td>CPU TIME: 38.214432</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MEMBER: Q42Q</td>
<td>CPU SU CONS: 32074</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>QW0231AC: 311849464</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TASK SEQ. NUMBER: 2</td>
<td>TASK TOKEN: X'6B9C57A0'</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TASK CREATION: 09/24/08 12:40:57.763105</td>
<td>TASK ELAPSED: 1:08.529116</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TASK TERMINATION: 09/24/08 12:42:06.294460</td>
<td>CPU TIME: 37.548378</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MEMBER: Q42Q</td>
<td>CPU SU CONS: 31515</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>QW0231AC: 311862600</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

STATEMENT NO

The statement number. If the PLAN_TABLE exists, this is the same as QUERYNO in the PLAN_TABLE.

Field Name: QW0231SN

QUERY BLOCK NO

The query block number. If the PLAN_TABLE exists, this is the same as QBLOCKNO in the PLAN_TABLE.

Field Name: QW0231QN

PARALLEL GROUP NO

The parallel group number. If the PLAN_TABLE exists, this is the same as ACCESS_PGROUP_ID in the PLAN_TABLE.

Field Name: QW0231GN

REPEAT.GRPS

The number of repeat groups in the section.

Field Name: QW0231RN

RECORD

The position of this record in the series of IFCID 222 records.

Field Name: QW0231TR

OF

The total number of IFCID 231 records in this series.

Field Name: QW0231NR

GROUP CREATION

The time of group creation in DB2 timestamp format.

Field Name: QW0231CT

GROUP TERMINATION

The time of group termination in DB2 timestamp format.
FIELD NAME: QW0231ET

GROUP ELAPSED
The elapsed time between group creation and group termination in DB2 timestamp format.
FIELD NAME: RT0231GE

TASK SEQ. NUMBER
The task sequence number.
FIELD NAME: QW0231TQ

TASK TOKEN
The task token.
FIELD NAME: QW0231TK

TASK CREATION
The time of task creation in DB2 timestamp format.
FIELD NAME: QW0231TC

TASK ELAPSED
The elapsed time between task creation and task termination in DB2 timestamp format. If this value is negative, N/C is printed.
FIELD NAME: RT0231EL

TASK TERMINATION
The time of task termination in DB2 timestamp format.
FIELD NAME: QW0231TT

CPU TIME
Task CPU execution time.
FIELD NAME: QW0231TX

MEMBER
The name of the DB2 member on which the task was executed.
FIELD NAME: QW0231TM

CPU SU CONS
The CPU service units that the task consumed.
FIELD NAME: QW0231SU
IFCID 233 - Call User Routine

This topic shows detailed information about “Record Trace - IFCID 233 - Call User Routine”.

IFCID 233 signals the start or end of a call to a user routine (stored procedure or user-defined function) at a DB2 server.

This record is written when performance trace class 3 is active. MONITOR PRIVILEGE is required for reading via IFI.

This record traces the caller's information.

Record trace - IFCID 233 - Call User Routine

The field labels shown in the following sample layout of “Record Trace - IFCID 233 - Call User Routine” are described in the following section.

LOCATION NAME
The location name.
Field Name: QW0233LN

COLLECTION ID
The package collection identifier.
Field Name: QW0233PC

PROGRAM NAME
The program name.
Field Name: QW0233PN

SCHEMA NAME
The name of the schema associated with this routine.
Field Name: QW0233SC

ROUTINE NAME
The specific name of the routine.
Field Name: QW0233PR

VERSION NAME
The name of the version.
Field Name: QW0233VER

ROUTINE TYPE
The routine type can have the following values:
IFCID 233 - Call User Routine

PROCEDURE
The routine is a stored procedure

FUNCTION
The routine is a User-Defined Function

Field Name: QW0233TY

CONSISTENCY TOKEN
The consistency token.

Field Name: QW0233TS

ENTRY/EXIT TYPE
The entry or exit event type can have the following values:

ENTERING
The agent is entering a routine.

RETURNED
The agent has returned from a routine.

Field Name: QW0233EX

NESTING LEVEL
The nesting level of the routine.

Field Name: QW0233NL

STATEMENT NO
The statement number of the statement executed.

Field Name: QW0233SN

TYPE
The statement type. Possible values are DYNAMIC or STATIC.

Field Name: QW0233STY

ROUTINE ID
The routine identifier.

Field Name: QW0233RID

STATEMENT ID
The unique identifier of the currently executing statement. It is shown as an integer and in hexadecimal format.

Field Name: QW0233SID

CONV INTO HEX
The unique identifier of the currently executing statement. It is shown as an integer and in hexadecimal format.

Field Name: QW0233SID
IFCID 234 - Calling Agent Auth IDs

This topic shows detailed information about “Record Trace - IFCID 234 - Calling Agent Auth IDs”.

This IFCID returns the authorization ID information for the calling agent.

Record trace - IFCID 234 - Calling Agent Auth IDs

The field labels shown in the following sample layout of “Record Trace - IFCID 234 - Calling Agent Auth IDs” are described in the following section.

**PRIM AUTHID:** USERID01

**SQL AUTHID:** SQLAUTHID01

**SEC AUTHID:** SQLAUTHID02

**SEC AUTHID:** SQLAUTHID03

**PRIM AUTHID**

The primary authorization ID.

*Field Name:* QW0234PN

**SQL AUTHID**

The SQL authorization ID.

*Field Name:* QW0234AN

**SEC AUTHID**

The secondary authorization ID.

*Field Name:* QW0234SN
IFCID 236 - DDF SNA XLN Protocol Error

This topic shows detailed information about “Record Trace - IFCID 236 - DDF SNA XLN Protocol Error”.

Record trace - IFCID 236 - DDF SNA XLN Protocol Error

The field labels shown in the following sample layout of “Record Trace - IFCID 236 - DDF SNA XLN Protocol Error” are described in the following section.

REMOTE LOCATION
The location name or LU name of the remote partner involved in the protocol error.

Field Name: QW0236LO

LAST OPERATION
Indicates whether the last network operation was a send or receive.

Field Name: QW0236SR

SENT
The last message sent by this DB2 site during the XLN exchange.

Field Name: QW0236MS

RCVD
The last message received by this DB2 site during the XLN exchange.

Field Name: QW0236MR

VTAM RPL
The VTAM RPL associated with the last XLN message received during the exchange log names (XLN).

Field Name: QW0236VR

EXT
The VTAM RPL extension which describes the LU 6.2 verb indicators for the last message received.

Field Name: QW0236VX
IFCID 237 - Set Current Degree

This topic shows detailed information about “Record Trace - IFCID 237 - Set Current Degree”.

This record is generated when an SQL SET CURRENT DEGREE statement is executed.

Record trace - IFCID 237 - Set Current Degree

The field labels shown in the following sample layout of “Record Trace - IFCID 237 - Set Current Degree” are described in the following section.

PREV DEGREE: 1  NEW DEGREE: ANY  STATUS: SUCCESSFUL

PREV DEGREE

The previous (current) degree.

Field Name: QW0237OI

NEW DEGREE

The new (attempted) degree.

Field Name: QW0237NI

STATUS

The status of the statement.

Field Name: QW0237ST
This topic shows detailed information about “Record Trace - IFCID 238 - IBM Service Record”.

This record is for IBM service use.
IFCID 239 - Overflow Package/DBRM

This record is produced when the agent has executed more than 10 packages.

- **IFCID 239 - Buffer Manager Accounting Data** on page 40-699
  This topic shows detailed information about “Record Trace - IFCID 239 - Buffer Manager Accounting Data”.

- **IFCID 239 - General Package Overflow Accounting Data** on page 40-703
  This topic shows detailed information about “Record Trace - IFCID 239 - General Package Overflow Accounting Data”.

- **IFCID 239 - Locking Data** on page 40-704
  This topic shows detailed information about “Record Trace - IFCID 239 - Locking Data”.

- **IFCID 239 - Package/DBRM Accounting Data** on page 40-708
  This topic shows detailed information about “Record Trace - IFCID 239 - Package/DBRM Accounting Data”.

- **IFCID 239 - RDS Package Accounting** on page 40-718
  This topic shows detailed information about “Record Trace - IFCID 239 - RDS Package Accounting”.

- **IFCID 239 - Resource Limit Facility** on page 40-720
  This topic shows detailed information about “Record Trace - IFCID 239 - Resource Limit Facility”.
IFCID 239 - Buffer Manager Accounting Data

Record trace - IFCID 239 - Buffer Manager Accounting Data

The field labels shown in the following sample layout of "Record Trace - IFCID 239 - Buffer Manager Accounting Data" are described in the following section.

**BUFFER MANAGER ACCOUNTING DATA**

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>BUFFER POOL ID</td>
<td></td>
</tr>
<tr>
<td>GETPAGES</td>
<td></td>
</tr>
<tr>
<td>GETPAGES FAILED</td>
<td></td>
</tr>
<tr>
<td>BUFFER UPDATES</td>
<td></td>
</tr>
<tr>
<td>SYNCHRON.WRTE</td>
<td></td>
</tr>
</tbody>
</table>

**BUFFER POOL ID**

The buffer pool ID used by this thread.

Field Name: QBACPID

**SYNCHRON. READ**

The number of synchronous read I/O operations. DB2 increments this counter for each media manager synchronous physical read. Asynchronous I/O requests are not counted.

Field Name: QBACRIO

**GETPAGES**

The number of Getpage requests. This counter is incremented by successful Getpage requests for queries processed in parallel for each thread and for all successful and unsuccessful Getpage requests for queries that are not processed in parallel.

**Background and Tuning Information**

Reducing the number of Getpages can improve DB2 performance by reducing the number of synchronous page reads. With fewer Getpages, the requested page is more likely to be returned from the buffer pool. CPU usage is also reduced.

Check the ratio of Getpages to SQL DML statements, as a rule of thumb, try and keep this ratio below six.

You might need to modify the database and query design, for example:

- Add indexes to tables to reduce the number of pages scanned.
- Reassess the number of tables used and denormalize them, if necessary.
  
  As an example, a large table with many columns can result in several pages being fetched to satisfy a simple query requesting just a few columns. Splitting such a table into several tables with fewer columns, tailored to queries, will result in fewer pages returned for each query.

- Use correlated rather than noncorrelated queries to force the use of an index.

Field Name: QBACGET

This is an exception field.

**SEQ. PREFETCH**
The number of SEQUENTIAL PREFETCH requests. This is incremented for each PREFETCH request. Each request can result in an I/O read. If it does, up to 32 pages can be read for SQL and up to 64 pages for utilities. For SQL, depending on the buffer pool size, a request does not result in an I/O if all the requested pages are already in the buffer pool.

DB2 can use sequential prefetch if the data is accessed in sequential order even though sequential prefetch was not requested at bind time. This is known as sequential detection and is not included in the sequential prefetch count. Sequential detection is included in dynamic prefetch requests field.

**Background and Tuning Information**

Table space scans and nonmatching index scans generally use sequential prefetch.

**Field Name:** QBACSEQ

This is an exception field.

**GETPAGES FAILED**

The number of times that a page requested for a query processed in parallel was unavailable because an I/O was in progress or the page was not found in the buffer pool. The agent does not wait, but control returns to the agent.

This counter is used only when queries are processed in parallel.

**Background and Tuning Information**

If this value is close to zero, most pages are already in the buffer pool, and wait time for synchronous I/O is small.

This counter can be high when, for example, there is a cluster index scan and the data is not truly clustered by the index key. In this instance, data pages are not accessed in their true order and the cluster ratio is not valid. Use the Runstats utility to update it.

The value of this field is also used to determine how many sequential prefetches of one page were scheduled.

**Field Name:** QBACNGT

**LIST PREFETCH**

The number of LIST PREFETCH requests.

**Special Considerations:**

1. List prefetch allows DB2 to access data pages efficiently even if the needed data pages are not contiguous. It can be used with single index access and is always used with multiple index access.
2. List prefetch is always used to access data from the inner table during a hybrid join.
3. Data pages are read in quantities equal to the sequential prefetch quantity, which depends on the buffer pool size and is usually 32 pages.
4. During bind time DB2 does not use list prefetch if the estimated number of RIDs to be processed would take more than 50% of the RID pool. During execution time, list prefetch processing terminates if DB2 detects that more than 25% of the rows in the table need to be accessed. If list prefetch is terminated, it is indicated in IFCID 125.
Field Name: QBACLPF
This is an exception field.

**HPOOL WRITES**

The number of successful requests issued by DB2 to synchronously move a page from the virtual buffer pool to the hiperpool.

This is the number of times that pages are cached in the hiperpool. Before reusing a buffer in a virtual pool for a page request, its old content needs to be saved in a hiperpool if it is a candidate for hiperpool caching. (Data accessed by parallel queries is not cached in a hiperpool.)

Field Name: QBACHWR
This is an exception field.

**BUFFER UPDATES**

The number of times a buffer update occurs. This is incremented every time a page is updated and is ready to be written to DASD. If the same page is updated twice, for example, the number is incremented by 2.

This number is kept for all types of pages including data pages and work-file pages.

**Background and Tuning Information**

A nonzero value indicates any of the following activities:
- SQL INSERT, UPDATE, or DELETE
- Merge scan join
- Internal sort activity on the work files

Check the access path to determine whether sort activity can be minimized or avoided.

Field Name: QBACSWS
This is an exception field.

**DYNAMIC PREFETCH**

The number of (dynamic) PREFETCH requests. This is triggered by sequential detection. This includes prefetches for segmented table spaces.

**Background and Tuning Information**

Dynamic prefetch is typically used for a SELECT or UPDATE that is run repeatedly, accessing the index for each access.

If sequential prefetch, list prefetch, and dynamic prefetch reads have large values, check whether the access path can be improved.

Field Name: QBACDPF
This is an exception field.

**HPOOL WRITES-FAILED**

The number of unsuccessful write requests because of a shortage of expanded storage. If this number is high, reduce the hiperpool size.

Field Name: QBACHWF
This is an exception field.

**SYNCHRON. WRITE**
IFCID 239 - Buffer Manager Accounting Data

The number of immediate (synchronous) write I/O operations.

**Background and Tuning Information**

Although an immediate write is rare, a small nonzero value is acceptable. A large value indicates that the system needs tuning.

**Field Name:** QBACIMW

This is an *exception* field.

**PAGES READ ASYN-PAR**

The number of asynchronous pages read by prefetch that the agent triggered.

**Background and Tuning Information**

This is used to determine the buffer pool hit ratio: (Getpage requests - Synchronous reads - Asynchronous pages read) / Getpage requests.

**Field Name:** QBACSIO

This is an *exception* field.

**HPOOL READS-FAILED**

The number of unsuccessful synchronous read requests. An unsuccessful read occurs when a requested page was found in the hiperpool, but its content was discarded by MVS. For hiperpools defined as CASTOUT=YES, the written data can be discarded by MVS if the hiperpool usage is low or if the expanded memory is not large enough to back the hiperpool.

If this number is large for CASTOUT=YES hiperpools, reduce the size of the hiperpool. For hiperpools defined as CASTOUT=NO, an unsuccessful read can only happen when the backing expanded storage page was explicitly reconfigured out of the system.

**Field Name:** QBACHRF

This is an *exception* field.
IFCID 239 - General Package Overflow Accounting Data

This topic shows detailed information about “Record Trace - IFCID 239 - General Package Overflow Accounting Data”.

Record trace - IFCID 239 - General Package Overflow Accounting Data

The field labels shown in the following sample layout of “Record Trace - IFCID 239 - General Package Overflow Accounting Data” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>QPKGPKGN</td>
<td>Number of packages</td>
</tr>
<tr>
<td>QPKGPKNF</td>
<td>Number of the first section</td>
</tr>
<tr>
<td>QPKGPKNL</td>
<td>Number of the last section</td>
</tr>
</tbody>
</table>

GENERAL PACKAGE OVERFLOW ACCOUNTING DATA
NUMBER OF PACKAGES 1 FIRST SECTION 2 LAST SECTION 3

NUMBER OF PACKAGES
The number of packages.
Field Name: QPKGPKGN

FIRST SECTION
The number of the first section in this record.
Field Name: QPKGPKNF

LAST SECTION
The number of the last section in this record.
Field Name: QPKGPKNL
IFCID 239 - Locking Data

IFCID 239 - Locking Data

This topic shows detailed information about “Record Trace - IFCID 239 - Locking Data”.

Record trace - IFCID 239 - Locking Data

The field labels shown in the following sample layout of “Record Trace - IFCID 239 - Locking Data” are described in the following section.

<table>
<thead>
<tr>
<th>LOCKING DATA</th>
<th>LOCK REQUEST</th>
<th>12 LOCK SUSPENSIONS</th>
<th>2 CLAIM REQUESTS</th>
<th>17</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEADLOCKS</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TIMEOUTS</td>
<td>3 UNLOCK REQUEST</td>
<td>13 DRUM LATCH SUSPENS.</td>
<td>11 CLAIM REQ. FAILED</td>
<td>18</td>
</tr>
<tr>
<td>ESCALATIONS(SHR)</td>
<td>4 QUERY REQUEST</td>
<td>14 OTHER SUSPENSIONS</td>
<td>8 DRAIN REQUESTS</td>
<td>19</td>
</tr>
<tr>
<td>ESCALATIONS(EXC)</td>
<td>5 CHANGE REQUEST</td>
<td>15 DRAIN REQ. FAILED</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAXIMUM PAGE/ROW LOCKS HELD</td>
<td>6 OTHER REQUEST</td>
<td>16</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

DEADLOCKS

The number of times deadlocks were detected. This number should be low, ideally 0.

Background and Tuning Information

Deadlocks occur when two or more application processes each hold locks on resources that the others need, without which they cannot proceed. Ensure that all applications accessing the same tables access them in the same order.

Deadlocks can also occur through index page splits if there is high insert activity. In this case, the recommendation is to set SUBPAGES to 1 for the index.

This field is incremented once for each deadlock encountered. There is no correlation between this field and the deadlock events reported in the Locking report set or the number of IFCID 172 records written. This field reports all deadlocks, regardless of how they were resolved. The locking report and record trace IFCID 172 show only those deadlocks that were resolved by DB2.

Field Name: QTXADEA

This is an exception field.

LOCK REQUEST

The number of requests to lock a resource.

Field Name: QTXALOCK

This is an exception field.

LOCK SUSPENSIONS

The number of times a lock could not be obtained and the unit of work was suspended.

Background and Tuning Information

This number should be low, ideally 0.

The number of lock suspensions is a function of the lock requests. Lock suspensions (or conflicts) can happen on either LOCK REQUEST or CHANGE REQUEST.

Suspensions are highly dependent on the application and table space locking protocols.

Field Name: QTXASLOC
This is an exception field.

CLAIM REQUESTS
The number of claim requests.
Field Name: QTXACLNO
This is an exception field.

TIMEOUTS
The number of times a unit of work was suspended for a time exceeding the timeout value. This number should be low, ideally 0.
Field Name: QTXATIM
This is an exception field.

UNLOCK REQUEST
The number of requests to unlock a resource.
This value can be less than the number of lock requests because DB2 can release several locks with a single unlock request.
Field Name: QTXAUNLK

IRLM LATCH SUSPENS.
The number of latch suspensions.
Field Name: QTXASLAT

CLAIM REQ. FAILED
The number of unsuccessful claim requests.
Field Name: QTXACLUN

ESCALATIONS(SHR)
The number of times the maximum page locks per table space are exceeded, and the table space lock escalates from a page lock (IS) to a table space lock (S) for this thread. You can specify the number of locks allowed per table space with the LOCKS PER TABLE(SPACE) parameter on the DB2 install panel DSNTIPJ.

Background and Tuning Information
Escalations can cause unpredictable response times. Lock escalations should only happen when an application process updates or references (if repeatable read is used) more pages than normal.
Field Name: QTXALES
This is an exception field.

QUERY REQUEST
The number of query requests.
Field Name: QTXAQR
This is an exception field.

OTHER SUSPENSIONS
The number of suspensions caused by something other than lock or latch.
Field Name: QTXASOTH
This is an exception field.

**DRAIN REQUESTS**

The number of drain requests.

**Field Name:** QTXADRNO

This is an exception field.

**ESCALATIONS(EXC)**

The number of times the maximum page locks per table space are exceeded and the table space lock escalates from a page lock (IX) to a table space lock (X).

**Background and Tuning Information**

Escalations can cause unpredictable response times. Lock escalations should only happen when an application process updates or references (if repeatable read is used) more pages than it normally does.

A useful rule of thumb is to compare the number of escalations (shared and exclusive) to the successful escalations (those that did not cause deadlocks and timeouts). If this value, or the number Lock escalations - shared and if the number of timeouts or deadlocks is also not 0, the timeout or deadlock is probably caused by the escalation.

If many escalations cause deadlocks and timeouts, the recommendation is to change the escalation threshold value. Use of ANY is extremely useful to prevent unnecessary and expensive page locks, for example locking all pages in a tablespace.

Lock escalations, shared or exclusive, should not be expected in a transaction environment.

**Field Name:** QTXALEX

This is an exception field.

**CHANGE REQUEST**

The number of change requests.

**Field Name:** QTXACHG

This is an exception field.

**DRAIN REQ. FAILED**

The number of unsuccessful drain requests.

**Field Name:** QTXADRUN

This is an exception field.

**MAXIMUM PAGE/ROW LOCKS HELD**

The maximum number of page or row locks concurrently held against all table spaces by a single application during its execution. This count is a high-water mark. It cannot exceed the LOCKS PER USER parameter on panel DSNTIPJ.

**Field Name:** QTXANPL

This is an exception field.

**OTHER REQUEST**
IFCID 239 - Locking Data

The number of requests to IRLM to perform a function other than LOCK, UNLOCK, QUERY, or CHANGE.

Field Name: QTXAIRLM

This is an exception field.
IFCID 239 - Package/DBRM Accounting Data

Record trace - IFCID 239 - Package/DBRM Accounting Data

The field labels shown in the following sample layout of “Record Trace - IFCID 239 - Package/DBRM Accounting Data” are described in the following section.

LOCATION

The location name.

If this field is blank in trace or report, the package or DBRM was executed locally. If it is not blank, all times represent the time spent locally to execute the remote package for this APPL_DIR requester.

This field is invalid (N/P) in case of DB2 10 or later if summary rollup data is present.

Field Name: QPACLOCN

This is an exception field.

COLLECTION

The package collection ID. This field does not apply to DBRMs. If the program name cannot be identified, this field is not present in report or trace.

This field is invalid in case of DB2 10 or later if summary rollup data is present. It can have the following value in:

- Accounting trace and report: N/P
- The Accounting FILE and SAVE PROGRAM table: blank

Field Name: QPACCOLN

This is an exception field.

PACKAGE ID

The program name (package ID or DBRM name).
In the case of rollup data (Accounting data of DDF/RRSAF threads and parallel tasks accumulated by DB2), the following value is shown:

- *ROLSUM* for DB2 10 or later
- *ROLLUP* for DB2 versions prior to DB2 10

**Field Name:** QPACPKID

This is an exception field.

**TOKEN**

The program (package or DBRM) consistency token.

This field is invalid (0) in case of DB2 10 or later if summary rollup data is present.

**Field Name:** QPACCONT

**SECTION NMB**

The number of this particular data section in the series.

**Field Name:** QPACRECN

**TYPE**

The program type. It can be DBRM (field name QPACDBRM) or package (field name QPACPACK).

**Field Name:** QPACFLGS

**SCHEMA NAME**

Schema name of the nested activity.

If the package is defined for a trigger, stored procedure, or user-defined function, then this field contains the name of the schema to which the nested activity belongs. It can have the following value in:

- Accounting Trace and Report: N/P
- The Accounting FILE and SAVE PROGRAM tables: blank

This field is invalid in case of DB2 10 or later if summary rollup data is present.

**Field Name:** QPACASCH

**SQL STMTS**

The number of SQL statements issued in this package or DBRM.

This number may not be equal to the total number of SQL statements in the QXST data section because QXST does not count all SQL statements. For example, it does not count commit or rollback statements.

**Note:** This field is shown for the following field labels in Accounting trace:

- SQL STMT - TOTAL
- SQL STMT - AVERAGE:
  - For DB2 9, the average is not calculated because it is identical to the TOTAL value. N/C (not calculated) is shown for this field.
  - For DB2 10 or later, the average data is shown.

**Field Name:** QPACSQLC

This is an exception field.
IFCID 239 - Package/DBRM Accounting Data

USED BY STOR.PROC
Indicates whether this package was loaded by a stored procedure.
This field is invalid in case of DB2 10 or later if unique or summary rollup data is present.
Field Name: QPACINSP

NON-ZERO CLASS 8
Indicates if Class 8 data is in this record.
Field Name: QPACCLS8

ACTIVITY NAME
The name of the nested activity.
This field contains the name of the nested activity if the package is defined for a:
• Trigger
• Stored procedure
• User-defined function (UDF)
• Native SQL procedure
• Non-inline UDF
In a data block that reports totals it is set to ALL NAMES.
This field is invalid in case of DB2 10 or later if summary rollup data is present.
It can have the following value in:
• Accounting Trace and Report: N/P
• The Accounting FILE and SAVE PROGRAM tables: blank
Field Name: QPACAANM

SUCC AUTH CHECK
Indicates whether a successful package EXECUTE authorization check was made and DB2 catalog access was avoided.
This field is invalid in case of DB2 10 or later if unique or summary rollup data is present.
Field Name: QPACPAC

LAST EXECUTED
This package or DBRM is either currently executing or is the most recently executed package or DBRM. This field is invalid in case of DB2 10 or later if unique or summary rollup data is present.
Field Name: QPACCRT

NON-ZERO CLASS 7
There is nonzero accounting class 7 data in this QPAC data instance.
Field Name: QPACCLS7

ACTIVITY TYPE
The type of activity. The following values indicate how the package was loaded:
ALL TYPES
   In a data block that reports totals it is set to ALL TYPES.

STORED PROC
   When running an external procedure

TRIGGER
   When running a trigger

UDF
   When running a user-defined function

NATIVE SQL PROC
   When running a native SQL procedure

NATIVE UDF
   When running a native UDF procedure (a non-inline user-defined function)

NONNESTED
   Indicates that none of the above values is true

MULTIPLE
   Indicates that packages with the same key but with different activity types were running

N/P
   Invalidated in case of rollup summary

The nested activity values that are shown in column NEST_ACTIVITY_TYPE of the table DB2PMFACCT_PROGRAM are:

S  For Stored Procedure
T  For Trigger
U  For UDF
Q  For native SQL procedure
D  For Native UDF
N  For nonnested (other)
blank  For invalidated in case of rollup summary

This field is invalid in case of DB2 10 or later if unique or summary rollup data is present.

Field Name: QPACAASFG

PACKAGE SWITCH

The number of times package was invoked from a different package. For the first package run by an application, the initial call counts as a package switch. If this package called a nested package (such as a trigger, UDF, or stored procedure), a switch will not be counted upon return from such a package.

Field Name: QPACSWITCH

ROLLED NBR THREADS

This value can be one of the following:

- In general, the number of threads to roll data into this QPAC data section. Non-rollup QPACs have a value of 1 and rollup QPACs have a value of 1 or more. With DB2 10 or later, this number is used as a divisor for calculating averages for package class 7, 8, or 10 times and events.
If ORDER (ACTNAME) is specified, the number of threads to roll data into this QPAC data section of a special activity type depends on the following:
- If IFCIDs 233 are available, the number of threads to roll data into this QPAC data section for a stored procedure (SP) or user-defined function (UDF). That number of subprograms called by these routines and functions is not taken into account.
- If IFCIDs 233 are not collected, the total number of threads to roll data into this QPAC data section. The sum also includes the number of subprograms.

Field Name: QPACRLNU
CLASS 7: BEGINNING STORE CLOCK TIME
The store clock time at entry to DB2 for the most recent execution of this package or DBRM.
This field is invalid in case of DB2 10 or later if unique or summary rollup data is present.
Field Name: QPACSCB
CLASS 7: ENDING STORE CLOCK TIME
The store clock time at exit from DB2 after the most recent execution of this package or DBRM.
This field is invalid in case of DB2 10 or later if unique or summary rollup data is present.
Field Name: QPACSCE
CLASS 7: BEGINNING TCB CPU TIME
The CPU time at entry to DB2 for the most recent execution of this package or DBRM. This time does not include the CPU time consumed on an IBM specialty engine.
This field is invalid in case of DB2 10 or later if unique or summary rollup data is present.
Field Name: QPACBJST
CLASS 7: ENDING TCB CPU TIME
The CPU time at exit from DB2 for the most recent execution of this package or DBRM. This time does not include CPU consumed on an IBM specialty engine.
This field is invalid in case of DB2 10 or later if unique or summary rollup data is present.
Field Name: QPACEJST
CLASS 7: TOTAL ELAPSED TIME
The total elapsed time for executing the package or DBRM.
Field Name: QPACSCT
CLASS 7: DB2 ENTRY/EXIT
The number of DB2 entries or exits processed during the execution of the package or DBRM.
In Accounting reports this is shown twice; as a total and as an average.
Field Name: QPACARNA

CLASS 7: TOTAL TCB TIME
The class 7 CPU time for all executions of the package or DBRM. This time does not include the:
• Class 7 time for parallel tasks
• CPU time that is consumed on an IBM specialty engine

Field Name: QPACTJST
This is an exception field.

CLASS 7: SE CPU TIME
The total CPU time for all executions of this package or DBRM that was consumed on an IBM specialty engine.

Note: All CPU times of an IBM Specialty Engine (SE) that are reported in DB2 trace records are already normalized by DB2 to the speed of the purpose processor.

Field Name: QPACCLS7_ZIIP

CLASS 8: LOCK/LATCH SUSP TIME
This field depends on the DB2 version that is installed:
• DB2 10 or later: The accumulated lock elapsed wait time that occurred while executing this package.
• Prior to DB2 10: The accumulated elapsed time spent by the package or DBRM waiting for lock and latch suspensions.

Background and Tuning Information
OMEGAMON XE for DB2 PE might adjust this value if the thread was suspended when performance data was gathered.
If the suspension time is high, investigate locking activity.

Field Name: QPACAWTL
This is an exception field.

CLASS 8: LOCK/LATCH SUSP EVENTS
This field depends on the DB2 version that is installed:
• DB2 10 or later: The number of wait trace events processed for waits for lock while executing this package.
• Prior to DB2 10: The number of wait trace events processed for lock or latch.

Field Name: QPACARNL

CLASS 8: WAIT TIME LOCAL LOCKS
The accumulated latch elapsed wait time for latch suspensions that occurred while executing this package.

Field Name: QPACAWLH

CLASS 8: LOCAL LOCK WAIT TRACE EVENTS
The number of wait trace events processed for page latch contention while executing this package.

Field Name: QPACANLH
CLASS 8: DB2 LATCH SUSP TIME
This field depends on the DB2 version that is installed:
- **DB2 10 or later:** The accumulated lock elapsed wait time that occurred while executing this package.
- **Prior to DB2 10:** The accumulated elapsed time spent by the package or DBRM waiting for lock and latch suspensions.

**Background and Tuning Information**
OMEGAMON XE for DB2 PE might adjust this value if the thread was suspended when performance data was gathered.
If the suspension time is high, investigate locking activity.

**Field Name:** QPACAWTL
This is an exception field.

CLASS 8: LATCH WAIT TRACE EVENTS
This field depends on the DB2 version that is installed:
- **DB2 10 or later:** The number of wait trace events processed for waits for lock while executing this package.
- **Prior to DB2 10:** The number of wait trace events processed for lock or latch.

**Field Name:** QPACARNL

CLASS 8: SYNCHRONOUS I/O SUSP TIME
The accumulated elapsed wait time for I/O suspensions under this thread during the execution of the package or DBRM.

**Background and Tuning Information**
OMEGAMON XE for DB2 PE might adjust this value if the thread was suspended when its performance data was gathered.

**Field Name:** QPACAWTI
This is an exception field.

CLASS 8: SYNCHRONOUS I/O SUSP EVENTS
The number of wait trace events processed for I/O.

**Field Name:** QPACARNE

CLASS 8: OTHER READ SUSP TIME
The accumulated waiting time for a read I/O performed under a thread other than this one during the execution of the package or DBRM.

**Background and Tuning Information**
OMEGAMON XE for DB2 PE might adjust this value if the thread was suspended when performance data was gathered.
This field includes waits caused by sequential prefetch, list prefetch, dynamic prefetch, and synchronous read I/O performed by other threads.
If the value in this field is high, the problem could be an I/O bound query using prefetch or an I/O contention. The application is accessing data from a busy data set, volume, or control unit and is continually being suspended. Consult the DBA and MVS systems programmer.
Field Name: QPACAWTR
This is an exception field.

CLASS 8: OTHER READ SUSP EVENTS
The number of suspensions due to read I/O.

Field Name: QPACARNR

CLASS 8: OTHER WRITE SUSP TIME
The accumulated waiting time due to a write I/O performed for another thread during the execution of a package or DBRM.

Background and Tuning Information
If the value in this field is high, the problem could be I/O contention. The application is accessing data from a busy data set, volume, or control unit and is continually being suspended. Consult the DBA and MVS systems programmer to resolve possible data set placement problems.

Field Name: QPACAWTW
This is an exception field.

CLASS 8: OTHER WRITE SUSP EVENTS
The number of suspensions due to write I/O.

Field Name: QPACARNW

CLASS 8: SERV.TASK SWITCH SUSP TIME
The accumulated waiting time due to a synchronous execution unit switch to DB2 services from this thread during the execution of the package or DBRM.

Background and Tuning Information
OMEGAMON XE for DB2 PE might adjust this value if the thread was suspended when its performance data was gathered.
This value includes the waits because of an OPEN/CLOSE data set, SYSLGRNG update, HSM RECALL data set, DATASPACE MANAGER services, DEFINE, EXTEND, and DELETE data set, and AUTONOMOUS PROCEDURE. Preformatting of data sets is a common cause of service task suspensions.

Field Name: QPACAWTE
This is an exception field.

CLASS 8: SERV.TASK SWITCH SUSP EVENTS
The number of wait trace events processed for DB2 service tasks.

Field Name: QPACARNS

CLASS 8: ARCH.LOG(QUIES) SUSP TIME
The accumulated waiting time caused by processing ARCHIVE LOG(QUIESCE) commands during the execution of the package or DBRM. This number represents the amount of time that an individual thread was suspended because of the command, not the time it took for the entire command to complete.

Background and Tuning Information
IFCID 239 - Package/DBRM Accounting Data

OMEGAMON XE for DB2 PE might adjust this value if the thread was suspended when its performance data was gathered.

Avoid issuing the -ARCHIVE LOG QUIESCE command during peak periods.

Field Name: QPACALOG

CLASS 8: ARCH. LOG (QUIES) SUSP EVENTS
The number of ARCHIVE LOG MODE (QUIESCE) commands issued.

Field Name: QPACALCT

CLASS 8: DRAIN LOCK SUSP TIME
The accumulated waiting time due to a drain lock.

Field Name: QPACAWDR

CLASS 8: DRAIN LOCK SUSP EVENTS
The number of wait trace events processed for waits for drain locks.

Field Name: QPACARND

CLASS 8: CLAIM RELEASE SUSP TIME
The accumulated waiting time for a drain waiting for claims to be released during the execution of the package or DBRM.

Background and Tuning Information
OMEGAMON XE for DB2 PE might adjust this value if the thread was suspended when its performance data was gathered.

Field Name: QPACAWCL

CLASS 8: CLAIM RELEASE SUSP EVENTS
The number of wait trace events processed for waits for claims to be released.

Field Name: QPACARNC

CLASS 8: PAGE LATCH SUSP TIME
The accumulated waiting time caused by a page latch contention.

Field Name: QPACAWTP

CLASS 8: PAGE LATCH SUSP EVENTS
The number of page latch wait trace events processed.

Field Name: QPACARNH

CLASS 8: NOTIFY MESSAGES SUSP TIME
The accumulated elapsed waiting time due to suspensions caused by sending notify messages to other members in the data sharing group. Messages are sent, for example, when database descriptors are changed due to DDL.

This value is only calculated if accounting class 8 is active and DB2 is a member of a DB2 data sharing group.

Field Name: QPACAWTG

CLASS 8: NOTIFY MESSAGES EVENTS
The number of wait trace events processed for sending notify messages to other members in the data sharing group.

Field Name: QPACARNG

CLASS 8: GLOBAL CONTENT. PARENT SUSP TIME
The accumulated wait time due to global contention for parent L-Locks. Parent L-Locks are any of the following L-Lock types: database, tablespace, table, or partition.

Field Name: QPACAWTJ

CLASS 8: GLOBAL CONTENT. PARENT EVENTS
The number of wait trace entry/exit events processed for waits for global lock contention for parent L-Locks.

Field Name: QPACARNJ

CLASS 8: UDF EXECUTED
The number of user-defined functions scheduled.

Field Name: QPACUDNU

CLASS 8: STORED PROCEDURE EXECUTED
The number of stored procedures scheduled.

Field Name: QPACSPNS

CLASS 8: TCP/IP LOB XML TIME
The number of wait trace events that were processed for waits for TCP/IP LOB and XML materialization while this package or DBRM was running.

Field Name: QPACALBC

CLASS 8: TCP/IP LOB XML EVENTS
The accumulated wait time for TCP/IP LOB and XML materialization while running this package or DBRM.

Field Name: QPACALBW

ACCELERATOR SUSP TIME
The accumulated wait time for requests to an accelerator while executing this package.

Field Name: QPACAACW

ACCELERATOR EVENTS
The number of wait trace events processed for requests to an accelerator while executing this package.

Field Name: QPACAACC
IFCID 239 - RDS Package Accounting

This topic shows detailed information about “Record Trace - IFCID 239 - RDS Package Accounting”.

Record trace - IFCID 239 - RDS Package Accounting

The field labels shown in the following sample layout of “Record Trace - IFCID 239 - RDS Package Accounting” are described in the following section.

<table>
<thead>
<tr>
<th>RDS PACKAGE ACCOUNTING</th>
<th>SELECTS</th>
<th>INSERTS</th>
<th>UPDATES</th>
<th>DELETES</th>
<th>DESCRIBES</th>
<th>PREPARES</th>
<th>OPENS</th>
<th>CLOSES</th>
<th>FETCHS</th>
<th>LOCK TABLES</th>
<th>SQL CALLS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>4</td>
<td>7</td>
<td>10</td>
<td>2</td>
<td>5</td>
<td>8</td>
<td>11</td>
<td>6</td>
<td>3</td>
<td>9</td>
</tr>
</tbody>
</table>

**SELECTS**

The number of SQL SELECT statements executed.

**Field Name**: QPSELECT

This is an *exception* field.

**INSERTS**

The number of INSERT statements executed.

**Field Name**: QPINSRT

This is an *exception* field.

**UPDATES**

The number of UPDATE statements executed.

**Field Name**: QPUPDTE

This is an *exception* field.

**DELETES**

The number of DELETE statements executed.

**Field Name**: QPDELETE

This is an *exception* field.

**DESCRIBES**

The number of data capture describes.

**Field Name**: QPDESC

**PREPARES**

The number of full prepare requests.

**Field Name**: QPPREP

**OPENS**

The number of full open requests.

**Field Name**: QPOPEN

**CLOSES**

The number of close requests.

**Field Name**: QPCLOSE

**FETCHS**

The number of fetch requests.
Field Name: QPFETCH
LOCK TABLES
The number of lock tables.
Field Name: QPLOCK
SQL CALLS
The number of SQL calls.
Field Name: QPCALL
IFCID 239 - Resource Limit Facility

IFCID 239 - Resource Limit Facility

This topic shows detailed information about “Record Trace - IFCID 239 - Resource Limit Facility”.

Record trace - IFCID 239 - Resource Limit Facility

The field labels shown in the following sample layout of “Record Trace - IFCID 239 - Resource Limit Facility” are described in the following section.

RES LIMIT SCOPE

Indicates how the resource limit was established. A value of 0 shows that the resource limit facility was not started.

Field Name: QTXAPREC

RLF TABLE ID

The identifier of the resource limit specification table.

Field Name: QTXARLID

LIMIT IN CPU 16 MICROSEC

The CPU time limit, in microseconds, set by the resource limit facility.

Field Name: QTXACLMT

RES LIMIT TYPE

Indicates how the type of resource limit was established: infinite, zero, or limit.

Note: Label QTXAFLG1 presents the first flag byte in hexadecimal:
X'80'  Infinite limit
X'40'  No run or zero limit

Field Name: QTXAFLG1

LIMIT IN SERVICE UNITS

The maximum number of CPU service units to be used. Normally, the value is not 0 if the RES LIMIT TYPE is LIMIT. A value of 0 indicates no limit.

Field Name: QTXASLMT

HIGHEST CPU 16 MICROSEC USED

The highest CPU time used by a single DB2 call, in microseconds. Note that there can be many DB2 calls for one SQL statement.

Field Name: QTXACHUS

QTXAFLG1 (S)

Indicates how the type of resource limit was established: infinite, zero, or limit.

Note: Label QTXAFLG1 presents the first flag byte in hexadecimal:
X'80'  Infinite limit
IFCID 239 - Resource Limit Facility

X'40'  No run or zero limit

Field Name: QTXAFLG1
IFCID 247 - SQLDA Data and Input Host Variable Data

This topic shows detailed information about “Record Trace - IFCID 247 - SQLDA Data and Input Host Variable Data”.

IFCID 247 records SQLDA data and INPUT HOST VARIABLE data are related to a user application program. Each host variable is traced individually as it is moved from the user application area to the DB2 address space.

For dynamic SQL statements of length 5000 or less, you can use these records in combination with records from IFCID 064 and IFCID 063 to determine which statements are associated with which host variables. To do this, you need to match the statement number in this record to the statement number in an IFCID 064 record. An IFCID 063 Record that follows the IFCID 064 record that has the same CORRELATION ID and ACE values contains the SQL statement associated with the host variables.

Record trace - IFCID 247 - SQLDA Data and Input Host Variable Data

The field labels shown in the following sample layout of “Record Trace - IFCID 247 - SQLDA Data and Input Host Variable Data” are described in the following section.

INPUT HOST VARIABLE TRACING
LOCATION NAME: PMO5D851
COLLECTION ID: ADBL410
PROGRAM NAME : ADB2REE
STATEMENT NUMBER : 2536  CONSISTENCY TOKEN : X'1725896E1B46AEB8'
LENGTH EACH SQLDA ENTRY: 12  NUMBER ENTRIES IN SQLDA: 1  FORMAT SQLDA : B'1000'

SQLDA ENTRY
SQLDA NAME: 'BLANK'  SQLDA ENTRY NUMBER: 1  DATA TYPE : 452  LENGTH: 18
ADDRESS TO DATA : X'1E44B692'  PRECISION (IF DEC): N/A  SCALE (IF DEC): N/A
ADDRESS TO NULL INDICATOR: X'FF000000'  NULL INDICATOR : NO

SQLDA DATA SECTION
LENGTH OF DATAREA: X'0012'
SQLDA DATA:
0000 4355252 454E5420 53455256 45522020 2020 | .....*---------

LOCATION NAME
Location name.

Field Name: QW0247LN

COLLECTION ID
Package collection identifier.

Field Name: QW0247PC

PROGRAM NAME
Program name.

Field Name: QW0247PN

STATEMENT NUMBER
Statement number.

Field Name: QW0247SN

CONSISTENCY TOKEN
The consistency token.

Field Name: QW0247TS
IFCID 247 - SQLDA Data and Input Host Variable Data

LENGTH EACH SQLDA ENTRY
Length of each SQLDA entry.
Field Name: QW0247LE

NUMBER ENTRIES IN SQLDA
Number of entries in the SQL data area.
Field Name: QW0247NE

FORMAT SQLDA
The format of the SQLDA. Possible values are:
0 - COMPRESSED
   Is a compressed form of the SQLDA.
1 - COMPLETE
   Is a complete SQLDA containing the data type, address, and
   address of the indicator variable for each host variable.
2 - FIXED LENGTH
   Is a variable length character format containing the length of the
   string and text.
? - UNKNOWN
   Is shown, if none of the above field names is used.
Field Name: QW0247FE

SQLDA NAME
SQLDA name, if Format 1 SQLDA. The first two bytes are the length of the
NAME and are not shown.
Field Name: QW0247NA

SQLDA ENTRY NUMBER
SQLDA entry number.
Field Name: QW0247NO

DATA TYPE
Is derived as described in DB2 SQL Reference, based on the SQLTYPE:
384, 385
   DATE
388, 389
   TIME
392, 393
   TIMESTAMP
448, 449
   VARYING LENGTH CHARACTER STRING
452, 453
   FIXED-LENGTH CHARACTER STRING
456, 457
   LONG VARYING CHARACTER STRING
480, 481
   FLOATING POINT
IFCID 247 - SQLDA Data and Input Host Variable Data

484, 485
PACKED DECIMAL

496, 497
LARGE INTEGER

500, 501
SMALL INTEGER

Note:
• Any other SQLTYPES are shown as: NON DISPLAYABLE DATA
• Values are shown in DB2 internal format.

Field Name: QW0247TY
LENGTH
Length of data for this entry. If the field type is decimal (484 or 485), the length is not applicable.
Field Name: QW0247LD
ADDRESS TO DATA
The address of the host variable in the application address space.
Field Name: QW0247PT
PRECISION (IF DEC)
If the field type is decimal (484 or 485), this is the precision.
Field Name: QW0247LP
SCALE (IF DEC)
If the field is decimal (484 or 485), this is the scale.
Field Name: QW0247LS
ADDRESS TO NULL INDICATOR
The address of the indicator variable, if the value in QW0247TY is odd (NULLABLE).
Field Name: QW0247IN
NULL INDICATOR
Null indicator values:
• YES, if X'00'
• NO, if X'FF'
Field Name: QW0247NL
LENGTH OF DATAREA
Field Name: QW0247LL
SQLDA DATA
Field Name: QW0247DA
This topic shows detailed information about “Record Trace - IFCID 248 - IBM Service Record”.

This record is for IBM service use.
This topic shows detailed information about “Record Trace - IFCID 249 - EDM Pool Invalidate DBD”.

This record traces DBD invalidations. A DBD is invalidated in the data sharing environment when one DB2 subsystem changes a DBD that needed, it is read, resulting in multiple copies of the DBD in the EDM pool.

**Record trace - IFCID 249 - EDM Pool Invalidate DBD**

The field labels shown in the following sample layout of “Record Trace - IFCID 249 - EDM Pool Invalidate DBD” are described in the following section.

- **DBID**: USIBMSYSTDB2
- **DATABASE NAME**: DSNDB01
- **DB2 MEMBER NAME**: AAAAAAAA

**DBID**

The database ID. Deduced from the DB2 fields QW0249ID, and QW0105DN or QW0107DN.

When present, the database name is shown, otherwise the decimal identifier from QW0249ID is shown or N/A when this value is 0.

**Field Name**: RT0249DB

**DATABASE NAME**

The database name.

**Field Name**: QW0249NM

**DB2 MEMBER NAME**

The name of the DB2 member causing the invalidation.

**Field Name**: QW0249MC
IFCID 250 - Connect/Rebuild Connect/Disconnect Group Bpool

This topic shows detailed information about “Record Trace - IFCID 250 - Connect/Rebuild Connect/Disconnect Group Bpool”.

This record is written for a group buffer pool (coupling facility cache structure) connect, rebuild, or disconnect event.

The DESCRIPTION column indicates what event occurred. The format of this record and data shown depends on the event being reported.

Record trace - IFCID 250 - Connect/Rebuild Connect/Disconnect Group Bpool

The field labels shown in the following sample layout of “Record Trace - IFCID 250 - Connect/Rebuild Connect/Disconnect Group Bpool” are described in the following section.

<table>
<thead>
<tr>
<th>STRUCTURE NAME: DSNCAT_GBP0</th>
<th>GROUP BP NAME: GBP0</th>
<th>RETURN CODE: 0</th>
<th>REASON CODE: X'00000000'</th>
</tr>
</thead>
<tbody>
<tr>
<td>STRUCTURE SIZE: 384</td>
<td>DIRECTORY ENTRIES: 1468</td>
<td>ALLOCATION: NO</td>
<td>MAX NUMBER CASTOUT: 1024</td>
</tr>
<tr>
<td>DATA ELEMENTS: 293</td>
<td>MAX STRUCTURE SIZE: 384</td>
<td>CONNECT TYPE: NEW CONNECTION</td>
<td>EXCLUSION LIST HONORED: YES</td>
</tr>
<tr>
<td>QW0250F1</td>
<td>B'00000000000000000000000000000000'</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

IFCID 250 - Connect & Rebuild Connect Failure (Authorization) Record

<table>
<thead>
<tr>
<th>STRUCTURE NAME: DSNCAT_GBP0</th>
<th>GROUP BP NAME: GBP0</th>
<th>RETURN CODE: 0</th>
<th>Reason Code: X'00000000'</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAF RETURN CODE: 9999999999</td>
<td>SAF REASON CODE: X'HHHHHHHH'</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

IFCID 250 - Connect & Rebuild Connect Failure (No Suitable Coupling Facility) Record

<table>
<thead>
<tr>
<th>STRUCTURE NAME: DSNCAT_GBP0</th>
<th>GROUP BP NAME: GBP0</th>
<th>RETURN CODE: 0</th>
<th>REASON CODE: X'00000000'</th>
</tr>
</thead>
<tbody>
<tr>
<td>COUPLING FACILITY NAME: XXXXXXX</td>
<td>COUPLING FACILITY REASON: STRUCTURE ATTRIBUTES INCONSISTENT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MIN CONTROL SPACE: 9999999999</td>
<td>TOTAL SPACE: 9999999999</td>
<td>TOTAL CONTROL SPACE: 9999999999</td>
<td>TOTAL FREE SPACE: 9999999999</td>
</tr>
<tr>
<td>FREE CONTROL SPACE: 9999999999</td>
<td>STORAGE SIZE: 999999</td>
<td>MAXIMUM ELEMENT: 999</td>
<td>MAXIMUM CASTOUT: 99999</td>
</tr>
</tbody>
</table>

IFCID 250 - Disconnect

<table>
<thead>
<tr>
<th>STRUCTURE NAME: DSNCAT_GBP0</th>
<th>GROUP BP NAME: GBP0</th>
<th>RETURN CODE: 0</th>
<th>REASON CODE: X'00000000'</th>
</tr>
</thead>
<tbody>
<tr>
<td>DISCONNECT TYPE: FAILED-PERSISTENT</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

STRUCTURE NAME

The name of the coupling facility structure.

Field Name: QW0250SN

GROUP BP NAME

The group buffer pool name.

Field Name: QW0250GN

RETURN CODE

The return code.

The reason code.

Field Name: QW0250RC

STRUCTURE SIZE
IFCID 250 - Connect/Rebuild Connect/Disconnect Group Bpool

The structure size, that is, the number of 4 KB blocks. The actual size might be less than the requested size due to insufficient space in the preference list facilities.
Field Name: QW0250SZ

DIRECTORY ENTRIES
The number of allocated directory entries.
Field Name: QW0250DR

ALLOCATION
Indicates whether this connect caused a structure allocation.
Field Name: QW0250CA

MAX NUMBER CASTOUT
The maximum number of castout classes.
Field Name: QW0250CO

DATA ELEMENTS
The number of data elements allocated. For DB2 group buffer pools, the size of the data elements is 4 KB. Each data entry consists of one or more data elements.
Field Name: QW0250DT

MAX STRUCTURE SIZE
The maximum structure size saved at the time the structure was allocated. The maximum structure size is obtained from the active policy at the time the structure is allocated.
Field Name: QW0250SM

CONNECT TYPE
The type of connection.
Field Name: QW0250CD

EXCLUSION LIST HONORED
Indicates whether the exclusion list was honored.
This field is only applicable if the value in ALLOCATION is YES. Otherwise, N/A is printed in this field.
Field Name: QW0250F2

SAF RETURN CODE
The system authorization facility (SAF) return code.
Field Name: QW0250X1

SAF REASON CODE
The SAF reason code.
Field Name: QW0250X2

COUPLING FACILITY NAME
The coupling facility name.
Field Name: QW0250ZN
IFCID 250 - Connect/Rebuild Connect/Disconnect Group Bpool

COUPLING FACILITY REASON
   The reason why the coupling facility was not suitable.
   Field Name: QW0250ZR

MIN CONTROL SPACE
   The minimum control space required (in 4 KB blocks) to allocate the
   structure for which connect was requested.
   Field Name: QW0250ZM

TOTAL SPACE
   The total space in the coupling facility in 4 KB blocks, including control
   and noncontrol space.
   Field Name: QW0250ZG

TOTAL CONTROL SPACE
   The total control space in the coupling facility in 4 KB blocks.
   Field Name: QW0250ZH

TOTAL FREE SPACE
   The total free space in 4 KB blocks, including control and noncontrol space.
   Field Name: QW0250ZI

FREE CONTROL SPACE
   The free control space in 4 KB blocks.
   Field Name: QW0250ZJ

STORAGE SIZE
   The storage increment size in 4 KB blocks.
   Field Name: QW0250ZK

MAXIMUM ELEMENT
   The maximum element characteristic. DB2 always requests a 4 KB element
   size. Therefore, this field should always equal 4.
   Field Name: QW0250ZL

MAXIMUM CASTOUT
   The maximum number of castout classes for a structure using this coupling
   facility.
   Field Name: QW0250ZO

DISCONNECT TYPE
   The type of disconnect.
   Field Name: QW0250DD
IFCID 251 - Buffer Manager PSET/Part P-Lock Request

This topic shows detailed information about “Record Trace - IFCID 251 - Buffer Manager PSET/Part P-Lock Request”.

Record trace - IFCID 251 - Buffer Manager PSET/Part P-Lock Request

The field labels shown in the following sample layout of “Record Trace - IFCID 251 - Buffer Manager PSET/Part P-Lock Request” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>P-LOCK TYPE</td>
<td>The P-lock type.</td>
</tr>
<tr>
<td>DBID</td>
<td>The database ID. Deduced from the DB2 fields QW0251KD, and QW0105DN or QW0107DN.</td>
</tr>
<tr>
<td>OBID</td>
<td>The object ID. Deduced from the DB2 fields QW0251KP, QW0105TN or QW0107TN.</td>
</tr>
<tr>
<td>PARTITION NMBR</td>
<td>The partition number. If this is a non-partitioned page set, 0 is printed in this field.</td>
</tr>
<tr>
<td>BP ID</td>
<td>The internal buffer pool ID (0-49 and 80-89).</td>
</tr>
<tr>
<td>IRLM FUNC CODE</td>
<td>The IRLM function code.</td>
</tr>
<tr>
<td>OBJECT TYPE</td>
<td>The object type.</td>
</tr>
</tbody>
</table>

P-LOCK TYPE

The P-lock type.

Field Name: QW0251KT

DBID

The database ID. Deduced from the DB2 fields QW0251KD, and QW0105DN or QW0107DN.

When present, the database name is shown, otherwise the decimal identifier from QW0251KD is shown or N/A when this value is 0.

Field Name: RT0251DB

OBID

The object ID. Deduced from the DB2 fields QW0251KP, QW0105TN or QW0107TN.

When present, the name of the object is shown, otherwise the decimal identifier from QW0251KP is shown or N/A if this value is 0.

Field Name: RT0251OB

PARTITION NMBR.

The partition number. If this is a non-partitioned page set, 0 is printed in this field.

Field Name: QW0251KR

BP ID

The internal buffer pool ID (0-49 and 80-89).

Field Name: QW0251KU

IRLM FUNC CODE

The IRLM function code.

Field Name: QW0251IF

OBJECT TYPE

The object type.

Field Name: QW0251OB
IFCID 251 - Buffer Manager PSET/Part P-Lock Request

REQUESTED STATE
The requested lock state if the value in IRLM FUNC CODE is LOCK or CHANGE. If the value is CHANGE FROM P-LOCK EXIT, then this is the P-lock state requested by the other member causing the P-lock exit of this member. In this case, this field is 0 if the request from the other member was not in conflict with the state of this member.

Field Name: QW0251ST

CONDITIONAL
Indicates whether the request was conditional.

Field Name: QW0251C1

RESTART
Indicates whether there was a restart lock request.

If the lock is currently retained on behalf of this DB2, a restart request causes the lock to be changed from retained to active. If the lock is not retained, the lock grant process is as normal.

Field Name: QW0251C6

MODIFY
Indicates whether this is a modify lock.

Field Name: QW0251C7

DATABASE NAME
The database name.

Field Name: QW0251DN

PAGESET NAME
The page set name.

Field Name: QW0251PN

DB2 MEMBER NAME
The DB2 member name that depends on the value in IRLM FUNC CODE:

- When CHANGE FROM P-LOCK EXIT this is the name of the database in conflict with the P-lock state currently held by this member.
- If it is not CHANGE FROM P-LOCK EXIT and the P-lock was rejected, this is the name of the database in conflict with this request

Field Name: QW0251DB

HELD STATE
Old and new P-lock held state. Old state taken from the DB2 field QW0251OS. New state taken from the DB2 field QW0251NS.

Field Name: RT251HS

CACHED STATE
Old and new P-lock cached state. Old state taken from the DB2 field QW0251OC. New state taken from the DB2 field QW0251NC.

Field Name: RT251CS
This topic shows detailed information about “Record Trace - IFCID 252 - IBM Service Record”.

This record is for IBM service use.
This topic shows detailed information about “Record Trace - IFCID 254 - Coupling Facility Cache Structure Statistics”.

Record trace - IFCID 254 - Coupling Facility Cache Structure Statistics

The field labels shown in the following sample layout of “Record Trace - IFCID 254 - Coupling Facility Cache Structure Statistics” are described in the following section.

GROUP BUFFER POOL NAME
The name of the group buffer pool.
Field Name: QW0254GN

EXPLICIT XI COUNTER
The number of times a request was made to the group coupling facility to explicitly cross invalidate a page and a number of XI signals were sent because the page was cached in one or more DB2 buffer pools.
Field Name: QW0254CI

READ HIT
The number of coupling facility read requests in which data was returned.
Field Name: QW0254RH

CHANGED PAGE WRITE HIT
The number of coupling facility write requests for changed pages that has successfully completed.
Field Name: QW0254WH

XI DIRECTORY ENTRY RECLAIM
The number of times that a directory entry was stolen and XI signals had to be sent because the page for the directory entry was cached in one or more DB2 buffer pools.
Field Name: QW0254XR

READ MISS DIRECTORY HIT
The number of coupling facility read requests for a page in which data was not returned but the page name was already assigned in the coupling facility directory (SES did not have to assign a directory entry for the page).
Field Name: QW0254RD

CLEAN PAGE WRITE HIT
The number of facility write requests for clean pages successfully completed.
IFCID 254 - Coupling Facility Cache Structure Statistics

Field Name: QW0254WC
CASTOUT
The number of castout operations performed.
Field Name: QW0254CC
READ MISS ASSIGNMENT SUPPRESSED
The number of times that a coupling facility read request specified a page for which no directory entry exists and no directory entry is created. DB2 does not create a directory entry if it does not need to register the page to the coupling facility for cross invalidation (XI); that is when no other DB2 member in the group has R/W interest in the page set/partition.
Field Name: QW0254RS
WRITE MISS CACHE FULL
The number of coupling facility write requests that could not complete due to a lack of coupling facility storage resources.
Field Name: QW0254WF
DIRECTORY ENTRY
The number of allocated directory entries (not cumulative).
Field Name: QW0254DE
READ MISS NAME ASSIGNED
The number of times that a coupling facility read request specified a page for which a directory entry was created.
Field Name: QW0254RN
DIRECTORY ENTRY RECLAIM
The number of times that a page name assignment required a coupling facility directory entry to be reclaimed (stolen).
Field Name: QW0254DR
DATA ENTRY
The number of allocated data entries (not cumulative).
Field Name: QW0254TE
READ MISS CACHE FULL
The number of times that a coupling facility read request specified a page for which no directory entry exists and no directory entry is created due to the lack of storage in the group buffer pool. A non-zero value in this field indicates that the backing coupling facility cache structure size might be too small to support the current workload.
Field Name: QW0254RF
DATA ENTRY RECLAIM
The number of times that a page name assignment required a coupling facility data entry to be reclaimed (stolen).
Field Name: QW0254TR
TOTAL CHANGED
IFCID 254 - Coupling Facility Cache Structure Statistics

The snapshot value of the current number of changed pages.

Field Name: QW0254TC

SEC-GBP CHANGED PAGE WRITE HIT

The number of successful coupling facility write requests for changed pages.

Field Name: QW02542W

SEC-GBP WRITE MISS CACHE FULL

The number of unsuccessful coupling facility write requests because of insufficient coupling facility storage resources.

Field Name: QW02542F

SEC-GBP DIRECTORY ENTRY

The number of allocated directory entries. This is a snapshot value.

Field Name: QW02542D

SEC-GBP DATA ENTRY

The number of allocated data entries. This is a snapshot value.
The number of allocated data entries that are currently in changed state. This is a snapshot value.

Field Name: QW02542C
IFCID 255 - Buffer Refresh Due to XI

This topic shows detailed information about “Record Trace - IFCID 255 - Buffer Refresh Due to XI”.

This record is written when a buffer refresh was caused by the cross invalidation (XI) of a data page in the group buffer pool. Cross invalidation occurs when a DB2 member of a data sharing group updates a data page and writes the newly changed page to the group buffer pool. All DB2 members that have this data page cached in their buffer pools are notified that the page was invalidated. If a member needs that data page, it must be refreshed.

Record trace - IFCID 255 - Buffer Refresh Due to XI

The field labels shown in the following sample layout of “Record Trace - IFCID 255 - Buffer Refresh Due to XI” are described in the following section.

```
DBID: NO PIECE NUMBER: X'00'
OBID: 4 PAGE NUMBER : X'000002'
BPID: 0 ACE TOKEN : N/P
TYPE: SYNCH FROM: GBPOOL

DBID
The database ID. Deduced from the DB2 fields QW0255DB, QW0105DN or QW0107DN.
When present, the database name is shown, otherwise the decimal identifier from QW0225DB is shown or N/A when this value is 0.
Field Name: RT0255DB

PIECE NUMBER
The data set number of the page set.
Field Name: QW0255PN

OBID
The object ID. Deduced from the DB2 fields QW0255OB, QW0105TN or QW0107TN.
When present, the name of the object is shown, otherwise the decimal identifier from QW0142OB is shown or N/A if this value is 0.
Field Name: RT0255OB

PAGE NUMBER
The relative page number within the data set.
Field Name: QW0255PG

BPID
The internal buffer pool ID.
Field Name: QW0255BP

ACE TOKEN
Ace token of the requester. This address ties the coupling facility read requests for prefetch to the allied agent or database access thread.
Field Name: QW0255AC

TYPE
Indicates whether the buffer refresh was synchronous or asynchronous.

**Field Name:** QW0255AS

FROM

Indicates whether data was returned from the group buffer pool or DASD.

**Field Name:** QW0255DR
IFCID 256 - Alter Group Buffer Pool

This topic shows detailed information about “Record Trace - IFCID 256 - Alter Group Buffer Pool”.

This record shows the old and the new status of the altered group buffer pool.

Record trace - IFCID 256 - Alter Group Buffer Pool

The field labels shown in the following sample layout of “Record Trace - IFCID 256 - Alter Group Buffer Pool” are described in the following section.

**BUFFER POOL GROUP BUFFER POOL ID: GBP0**

<table>
<thead>
<tr>
<th>OLD</th>
<th>NEW</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>

**GROUP BUFFER POOL ID**

The DB2 group buffer pool ID.

*Field Name:* QW0256GB

**DIRECTORY TO DATA RATIO**

The directory entry to data entry ratio. This is the value specified in the RATIO keyword of the ALTER GROUPBUFFERPOOL command.

New status deduced from the DB2 field QW0256NR.

Old status deduced from the DB2 field QW0256OR.

*Field Name:* RT0256DR

**CLASS CASTOUT THRESHOLD (%)**

The threshold at which the class castout is to be initiated. It is expressed as a percentage of the group buffer pool size. This is the value specified in the CLASST keyword of the ALTER GROUPBUFFERPOOL command.

New status deduced from the DB2 field QW0256NC.

Old status deduced from the DB2 field QW0256OC.

*Field Name:* RT0256CT

**CLASS CASTOUT THRESHOLD (PAGES) (OLD)**

The old class castout threshold based on the number of pages.

*Field Name:* QW0256ON

**CLASS CASTOUT THRESHOLD (PAGES) (NEW)**

The new class castout threshold based on the number of pages.

*Field Name:* QW0256NN

**GBP CASTOUT THRESHOLD (%)**

The threshold at which the castout is to be initiated for the group buffer pool. This is the value specified in the GBPOOLT keyword of the ALTER GROUPBUFFERPOOL command.
New status deduced from the DB2 field QW0256NG.
Old status deduced from the DB2 field QW0256OG.

Field Name: RT0256GT

GBP CHECKPOINT INTERVAL (MIN)
The time interval (in minutes) between successive group buffer pool checkpoints. This is the value specified in the GBPCHKPT keyword of the ALTER GROUPBUFFERPOOL command.

New status deduced from the DB2 field QW0256NK.
Old status deduced from the DB2 field QW0256OK.

Field Name: RT0256CI

GBP CACHE SETTING
GBPCACHE value before and after the ALTER GROUPBUFFERPOOL command was issued. This field specifies whether DB2 should write changed pages for the group buffer pool dependant pageset or partitions directly to DASD and use the group buffer pool only for sending XI signals.

New status deduced from the DB2 field QW0256NB.
Old status deduced from the DB2 field QW0256OB.

Field Name: RT0256CS

AUTOREC
A flag indicating how the AUTOREC option of the ALTER GROUPBUFFERPOOL command has been set. It specifies whether DB2 should automatically recover if GBP fails. The old value specifies the AUTOREC value before the ALTER GBP command was issued. The new value specifies the AUTOREC value after the ALTER GBP command was issued.

New status deduced from the DB2 field QW0256NA.
Old status deduced from the DB2 field QW0256OA.

Field Name: RT0256AR
This topic shows detailed information about “Record Trace - IFCID 257 - IRLM Notify Req Detail”.

This record shows the inter-DB2 notify message sending detail. IRLM notify requests are used to communicate among members of a DB2 data sharing group.

**Record trace - IFCID 257 - IRLM Notify Req Detail**

The field labels shown in the following sample layout of “Record Trace - IFCID 257 - IRLM Notify Req Detail” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOCK RES TYPE</td>
<td>The locked resource type. <strong>Note:</strong> For data sharing, SKELETON CURSOR TABLE LOCKING and SKELETON PACKAGE TABLE LOCK are LP-locks (an LP-lock has an L-lock component and a P-lock component).</td>
</tr>
<tr>
<td>DBID</td>
<td>The database ID. This field is not applicable if the value in LOCK RES TYPE is: SKELETON CURSOR TABLE LOCKING, UTILITY Serialization lock, SKELETON PACKAGE TABLE LOCK, COLLECTION, BINDLOCK, ALTER BUFFER POOL, GROUP BUFFERPOOL START/STOP LOCK, GROUP BUFFER POOL LEV CASTOUT P-LOCK, CATMAINT MIGRATION LOCK, CATMAINT CONVERT CATALOG LOCK, CATMAINT CONVERT DIRECTORY LOCK</td>
</tr>
<tr>
<td>OBID</td>
<td>The object ID. This field is not applicable if the value in LOCK RES TYPE is: SKELETON CURSOR TABLE LOCKING, UTILITY Serialization lock, SKELETON PACKAGE TABLE LOCK, COLLECTION, BINDLOCK, ALTER BUFFER POOL, GROUP BUFFERPOOL START/STOP LOCK, GROUP BUFFER POOL LEV CASTOUT P-LOCK, CATMAINT MIGRATION LOCK, CATMAINT CONVERT CATALOG LOCK, CATMAINT CONVERT DIRECTORY LOCK</td>
</tr>
<tr>
<td>Field Name</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
</tr>
<tr>
<td>RESOURCE ID</td>
<td>The hexadecimal identifier of the small resource. If LOCK RES TYPE is:</td>
</tr>
<tr>
<td><strong>DATA PAGE LOCKING</strong></td>
<td>First 3 bytes are the page number</td>
</tr>
<tr>
<td><strong>PARTITION LOCKING</strong></td>
<td>Last byte is the partition number</td>
</tr>
<tr>
<td><strong>INDEX PAGE LOCKING</strong></td>
<td>First 3 bytes are the page number and the last byte is the subpage number</td>
</tr>
<tr>
<td><strong>HASH ANCHOR LOCK</strong></td>
<td>First 3 bytes are the page number and the last byte is the anchor point ID</td>
</tr>
<tr>
<td><strong>CS-READ DRAIN</strong></td>
<td>Last byte is the partition number (optional)</td>
</tr>
<tr>
<td><strong>RR-READ DRAIN</strong></td>
<td>Last byte is the partition number (optional)</td>
</tr>
<tr>
<td><strong>WRITE DRAIN</strong></td>
<td>Last byte is the partition number (optional)</td>
</tr>
<tr>
<td><strong>ROW LOCK</strong></td>
<td>First 3 bytes are the page number and the last byte is the row ID of the record</td>
</tr>
<tr>
<td><strong>INDEX END OF FILE LOCK</strong></td>
<td>Last byte is the partition number (optional)</td>
</tr>
<tr>
<td><strong>PAGESET/PARTITION P-LOCK</strong></td>
<td>First byte is the 1-based partition number (optional)</td>
</tr>
<tr>
<td><strong>PAGE P-LOCK</strong></td>
<td>First byte is the 1-based partition number (optional) and the last 3 bytes are the relative page number</td>
</tr>
<tr>
<td><strong>PAGESET/PARTITION LEV CASTOUT P-LOCK</strong></td>
<td>First byte is the 1-based partition number (optional)</td>
</tr>
</tbody>
</table>

**Note:** In large partitioned table spaces, the page number covers 4 bytes instead of 3.

For all other lock resource types, the resource ID is not applicable.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>STATE</td>
<td>The lock state. This field is only applicable if the value in OPERATION is SEND. Otherwise, N/A is printed in this field.</td>
</tr>
</tbody>
</table>

**FIELD NAME:** QW0021KR

**NUMBER OF HOLDERS**
IFCID 257 - IRLM Notify Req Detail

The number of lock holders notified. This field is only applicable if the value in OPERATION is SEND. Otherwise, N/A is printed in this field.

Field Name: QW0257NU

OPERATION

The notify operation.

Field Name: QW0257OP

REQUEST

Indicates whether the request was synchronous or asynchronous. This field is only applicable if the value in OPERATION is SEND. Otherwise, N/A is printed in this field.

Field Name: QW0257FL
IFCID 258 - Data Set Extend Activity

This topic shows detailed information about “Record Trace - IFCID 258 - Data Set Extend Activity”.

This record is written every time a data set is extended.

Record trace - IFCID 258 - Data Set Extend Activity

The field labels shown in the following sample layout of “Record Trace - IFCID 258 - Data Set Extend Activity” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATA SET NAME</td>
<td>Data set name.</td>
</tr>
<tr>
<td>DATABASE NAME</td>
<td>Database name.</td>
</tr>
<tr>
<td>TABLESPACE NAME</td>
<td>Table or index space name.</td>
</tr>
<tr>
<td>PSID</td>
<td>Page set identifier.</td>
</tr>
<tr>
<td>PRIMARY QUANTITY</td>
<td>Primary allocation quantity in 4 KB units.</td>
</tr>
<tr>
<td>SEC. QUANTITY</td>
<td>Secondary allocation quantity in 4 KB units.</td>
</tr>
<tr>
<td>HIGH ALLOC BEFORE</td>
<td>High allocated space before the extend in 4KB units.</td>
</tr>
</tbody>
</table>

Chapter 40. IFCID Record Blocks 40-743
IFCID 258 - Data Set Extend Activity

Field Name: QW0258HB
HIGH ALLOC AFTER
High allocated space after the extend in 4 KB units.
Field Name: QW0258HA
MAX DS SIZE
Maximum size for the data set in 4 KB units.
Field Name: QW0258MS
EXTENTS BEFORE
Number of extents before the reported extend.
Field Name: QW0258XB
EXTENTS AFTER
Number of extents after the reported extend.
Field Name: QW0258XA
MAX EXTENTS
The maximum number of extents for the VSAM data set.
Field Name: QW0258XM
VOLUMES BEFORE
Number of volumes before the extend.
Field Name: QW0258VB
VOLUMES AFTER
Number of volumes after the extend.
Field Name: QW0258VA
MAX VOLUMES
Maximum number of volumes in the VSAM data set.
Field Name: QW0258VM
IFCID 259 - Buffer Manager Pg P-Lock Req

This topic shows detailed information about “Record Trace - IFCID 259 - Buffer Manager Pg P-Lock Req”.

Record trace - IFCID 259 - Buffer Manager Pg P-Lock Req

The field labels shown in the following sample layout of “Record Trace - IFCID 259 - Buffer Manager Pg P-Lock Req” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>P-LOCK TYPE</td>
<td>The P-lock type. This field can only have one value: PAGE.</td>
</tr>
<tr>
<td>DBID</td>
<td>The database ID. Deduced from the DB2 fields QW0259KD, QW0105DN or QW0107DN.</td>
</tr>
<tr>
<td>OBID</td>
<td>The database ID. Deduced from the DB2 fields QW0259KP, QW0105TN or QW0107TN.</td>
</tr>
<tr>
<td>PARTITION NMBR</td>
<td>The partition number. If this is a nonpartitioned page set, 0 is printed in</td>
</tr>
<tr>
<td>BP ID</td>
<td>The internal buffer pool ID (0-49 and 80-89).</td>
</tr>
<tr>
<td>IRLM FUNC CODE</td>
<td>The IRLM function code.</td>
</tr>
<tr>
<td>OBJECT TYPE</td>
<td>The object type.</td>
</tr>
<tr>
<td>MODIFY</td>
<td>Indicates whether this is a modify lock.</td>
</tr>
</tbody>
</table>

Chapter 40. IFCID Record Blocks  40-745
IFCID 259 - Buffer Manager Pg P-Lock Req

Field Name: QW0259C7

PAGE NMBR
The relative page number.
Field Name: QW0259KQ

DB2 MEMBER NAME
The DB2 member name that depends on the value in IRLM FUNC CODE:
• When CHANGE FROM P-LOCK EXIT this is the name of the database in conflict with the P-lock state currently held by this member.
• If it is not CHANGE FROM P-LOCK EXIT and the P-lock was rejected, this is the name of the database in conflict with this request

Field Name: QW0259DB

CONDITIONAL
Indicates whether the request was conditional.
Field Name: QW0259C1

RESTART
Indicates whether there was a restart lock request.
If the lock is currently retained on behalf of this DB2, a restart request causes the lock to be changed from retained to active. If the lock is not retained, the lock grant process is as normal.
Field Name: QW0259C6

REQUESTED STATE
The requested lock state if the value in IRLM FUNC CODE is LOCK or CHANGE. If the value is CHANGE FROM P-LOCK EXIT, then this is the P-lock state requested by the other member causing the P-lock exit of this member.
Field Name: QW0259ST

OLD HELD STATE
The previously held P-LOCK state.
Field Name: QW0259PS

NEW HELD STATE
The newly held P-LOCK state.
Field Name: QW0259NS
This topic shows detailed information about “Record Trace - IFCID 260 - IBM Service Record”.

This record is for IBM service use.
IFCID 261 - Group Buffer Pool Checkpoint

Record trace - IFCID 261 - Group Buffer Pool Checkpoint

The field labels shown in the following sample layout of “Record Trace - IFCID 261 - Group Buffer Pool Checkpoint” are described in the following section.

<table>
<thead>
<tr>
<th>BUFFERPOOL ID</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>CASTOUT P-LOCKS</td>
<td>0</td>
</tr>
<tr>
<td>INIT BY SPECIAL CASTOUT</td>
<td>0</td>
</tr>
<tr>
<td>INIT W/O SENDING MSG</td>
<td>0</td>
</tr>
<tr>
<td>INIT BY SENDING MSG</td>
<td>0</td>
</tr>
</tbody>
</table>

**BUFFERPOOL ID**

The internal identifier of the buffer pool. The values 0 through 49 are the identifiers for BP0 through BP49. The values 80 through 89 are the identifiers for BP32K through BP32K9.

**Field Name:** QW0261BP

**CASTOUT P-LOCKS**

The number of page sets or partition castout P-locks obtained by the GBP checkpoint process.

**Field Name:** QW0261PD

**NEW RECOVERY LRSN**

The global recovery record sequence number (LRSN) for this GBP checkpoint.

**Field Name:** QW0261NL

**START TIME**

The date and time at which GBP checkpoint processing started. The RECORD TIME field shows when processing ended.

**Field Name:** QW0261TS

**INIT BY SPECIAL CASTOUT**

The number of page sets and partitions for which a castout had to be initiated by a special castout process because the castout owner did not exist for the page set or partition.

**Field Name:** QW0261PS

**OLD RECOVERY LRSN**

The global recovery log record sequence number (LRSN) of the GBP checkpoint prior to this one.

**Field Name:** QW0261OL

**ELAPSED TIME**

The duration of the GBP checkpoint process. Calculated by QW0261TS - QWHSSTCK.

**Field Name:** RT0261ET

**INIT W/O SENDING MSG**
IFCID 261 - Group Buffer Pool Checkpoint

The number of page sets and partitions for which a castout was locally initiated without a message being sent.

Field Name: QW0261PL

NEW MINIMUM LRSN

The minimum restart/redo point for this GBP checkpoint.

Field Name: QW0261NM

READ DIRECTORY INFO

The number of coupling facility requests to read directory information.

Field Name: QW0261RD

INIT BY SENDING MSG

The number of page sets and partitions for which a castout was initiated by sending a message to the castout owner.

Field Name: QW0261PN

OLD MINIMUM LRSN

The minimum restart/redo point of the GBP checkpoint prior to this one.

Field Name: QW0261OM

DIRECTORY ENTRIES

The number of directory entries for changed pages processed.

Field Name: QW0261DP
IFCID 262 - GBPOOLT Castout Threshold Processing

This topic shows detailed information about “Record Trace - IFCID 262 - GBPOOLT Castout Threshold Processing”.

GBPOOLT castout threshold processing shows the data from IFCID 262. This IFCID contains statistics related to the GBPOOLT castout threshold processing for a GBP. It is only written if the GBPOOLT threshold has been reached.

This record is only written in a data sharing environment.

Record trace - IFCID 262 - GBPOOLT Castout Threshold Processing

The field labels shown in the following sample layout of “Record Trace - IFCID 262 - GBPOOLT Castout Threshold Processing” are described in the following section.

<table>
<thead>
<tr>
<th>Field Label</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>BUFFERPOOL ID</td>
<td>QW0262BP</td>
</tr>
<tr>
<td>READ CASTOUT CLASS</td>
<td>QW0262RC</td>
</tr>
<tr>
<td>CASTOUT P-LOCKS</td>
<td>QW0262PD</td>
</tr>
<tr>
<td>START TIME</td>
<td>QW0262TS</td>
</tr>
<tr>
<td>READ CASTOUT STATISTICS</td>
<td>QW0262RS</td>
</tr>
<tr>
<td>CHANGED PAGES CASTOUT</td>
<td>QW0262CP</td>
</tr>
<tr>
<td>ELAPSED TIME</td>
<td>QW0262TS</td>
</tr>
<tr>
<td>INIT BY SENDING MSG</td>
<td>QW0262BP</td>
</tr>
<tr>
<td>INIT W/O SENDING MSG</td>
<td>QW0262BP</td>
</tr>
</tbody>
</table>

BUFFERPOOL ID

The internal identifier of the buffer pool. The values 0 through 49 are the identifiers for BP0 through BP49. The values 80 through 89 are the identifiers for BP32K through BP32K9.

Field Name: QW0262BP

READ CASTOUT CLASS

The number of coupling facility read castout class requests.

Field Name: QW0262RC

CASTOUT P-LOCKS

The number of page set or partitions castout P-locks obtained by the GBPOOLT process.

Field Name: QW0262PD

START TIME

The date and time at which the GBP castout started. The RECORD TIME field shows when processing ended.

Field Name: QW0262TS

READ CASTOUT STATISTICS

The number of coupling facility requests to cast out statistics.

Field Name: QW0262RS

CHANGED PAGES CASTOUT

The number of changed page names that were passed to the page set and partition castout owner for castout.

Field Name: QW0262CP

ELAPSED TIME

The duration of the GBP castout process. Calculated by QW0262TS - QWHSSSTCK.
Field Name: RT0262ET
INIT BY SENDING MSG
The number of page sets and partitions for which castout was initiated by sending a notify message to the castout owner.
Field Name: QW0262PN
CHANGED PAGES IN GBP
The number of changed pages in group buffer pool.
Field Name: QW0262DP
FIRST CASTOUT CLASS
The first castout class processed.
Field Name: QW0262FC
INIT W/O SENDING MSG
The number of changed pages and partitions for which castout was locally initiated without a message being sent to the castout owner.
Field Name: QW0262PL
CHANGED PAGES GBPOOLT
The number of changed pages required to reach the GBPOOLT.
Field Name: QW0262GT
LAST CASTOUT CLASS
The last castout class processed. Sometimes the value in this field is smaller than the one in the FIRST CASTOUT CLASS field. This can happen if DB2 wraps around at the end of the castout class numbers.
Field Name: QW0262LC
This topic shows detailed information about “Record Trace - IFCID 263 - Page Set and Partition Castout Detail”.

This record shows page set and partition castout statistics. It is written by the page set or partition castout owner after the castout engine completed servicing the castout request.

This record is only written in a data sharing environment.

**Record trace - IFCID 263 - Page Set and Partition Castout Detail**

The field labels shown in the following sample layout of “Record Trace - IFCID 263 - Page Set and Partition Castout Detail” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BUFFERPOOL ID</td>
<td>The internal identifier of the buffer pool. The values 0 through 49 are the identifiers for BP0 through BP49. The values 80 through 89 are the identifiers for BP32K through BP32K9.</td>
</tr>
<tr>
<td>CASTOUT REASON</td>
<td>The reason why the castout was invoked.</td>
</tr>
<tr>
<td>DATABASE ID</td>
<td>The ID of the database.</td>
</tr>
<tr>
<td>PAGE SET OBJECT ID</td>
<td>The ID of the page set object.</td>
</tr>
<tr>
<td>PARTITION NUMBER</td>
<td>The partition number. It is 0 if this is a non-partitioned page set.</td>
</tr>
<tr>
<td>START TIME</td>
<td>The date and time at which castout processing started.</td>
</tr>
<tr>
<td>CASTOUT DATA REQUESTS</td>
<td>The number of castout requests.</td>
</tr>
<tr>
<td>UNLOCK FOR CASTOUT</td>
<td>The number of locks acquired for the castout.</td>
</tr>
<tr>
<td>READ CASTOUT CLASS</td>
<td>The class of the read castout.</td>
</tr>
<tr>
<td>DELETE NAME REQUESTS</td>
<td>The number of delete name requests.</td>
</tr>
<tr>
<td>WRITE I/O REQUESTS</td>
<td>The number of write I/O requests.</td>
</tr>
<tr>
<td>SEC-GBP DEL NAME LIST</td>
<td>The number of SEC-GBP del name list requests.</td>
</tr>
<tr>
<td>TIME DEL-NAME GBP</td>
<td>The time to process the SEC-GBP del name requests.</td>
</tr>
<tr>
<td>QW0263FL</td>
<td>The internal identifier of the SEC-GBP del name.</td>
</tr>
<tr>
<td>QW0263S1</td>
<td>The internal identifier of the SEC-GBP del name list.</td>
</tr>
<tr>
<td>QW0263S2</td>
<td>The internal identifier of the SEC-GBP del name list.</td>
</tr>
</tbody>
</table>

**BUFFERPOOL ID**

The internal identifier of the buffer pool. The values 0 through 49 are the identifiers for BP0 through BP49. The values 80 through 89 are the identifiers for BP32K through BP32K9.

Field Name: QW0263BP

**CASTOUT REASON**

The reason why the castout was invoked.

- CLASS THRESHOLD
- GROUP BUFFER POOL THRESHOLD
- GROUP BUFFER POOL CHECKPOINT
- GROUP BUFFER REBUILD
- SYNCHRONOUS CASTOUT
- CONVERTED TO NON-GBP-DEPENDENT OR ASYNCH. CASTOUT

Field Name: QW0263RS

**DATABASE ID**

The ID of the database.

Field Name: QW0263DB

**PAGE SET OBJECT ID**

The ID of the page set object.

Field Name: QW0263PS

**START TIME**

The date and time at which castout processing started.

Field Name: QW0263TS

**PARTITION NUMBER**

The partition number. It is 0 if this is a non-partitioned page set.
PRIVATE BUFFER
The number of private buffer allocated to this engine (in 4K increments).
Field Name: QW0263PB

ELAPSED TIME
The duration of the castout process. The RECORD TIME field shows when this process ended. Calculated by QW0263TS - QWHSSSTCK.
Field Name: RT0263ET

CASTOUT DATA REQUESTS
The number of coupling facility requests to cast out data.
Field Name: QW0263CD

UNLOCK FOR CASTOUT
The number of coupling facility requests to unlock for a castout.
Field Name: QW0263UN

READ CASTOUT CLASS
The number of coupling facility requests to read a castout class.
Field Name: QW0263RC

DELETE NAME REQUESTS
The number of coupling facility requests to delete a name.
Field Name: QW0263DN

WRITE I/O REQUESTS
The number of write I/O requests.
Field Name: QW0263IO

GBP DEL NAME LIST
The number of IXLCACHE delete_name_list requests to the secondary group buffer pool when the GBP MODE is DUPLEX.
Field Name: QW02632D

TIME DEL-NAME GBP
Duration of DELETE_NAME to primary GBP.
Field Name: QW02632T

DEL-NAME GBP
The number of times a DELETE_NAME request was reissued to the primary GBP.
Field Name: QW0263RD

TIME DEL-NAME SEC-GBP
Duration of DELETE_NAME to the secondary GBP.
Field Name: QW02632T

DEL-NAME SEC-GBP
The number of times a DELETE_NAME request was reissued to the secondary GBP for duplexing.

Field Name: QW02632R
This topic shows detailed information about “Record Trace - IFCID 265 - IBM Service Record”.

This record is for IBM service use.
This topic shows detailed information about “Record Trace - IFCID 266 - IBM Service Record”.

This record is for IBM service use.
This topic shows detailed information about “Record Trace - IFCID 267 - CF Rebuild/Alter/Start”.

This record shows the start of a coupling facility (CF) rebuild or alter, which is indicated by the OPERATION field. A rebuild or alter is reported in the same format.

**Record trace - IFCID 267 - CF Rebuild/Alter/Start**

The field labels shown in the following sample layout of “Record Trace - IFCID 267 - CF Rebuild/Alter/Start” are described in the following section.

**OPERATION**

The operation for the DB2 data sharing coupling facility structures:

- **F**: The rebuild due to the coupling facility structure failure or loss of connectivity to the coupling facility.
- **O**: The MVS rebuild initiated by the MVS operator command SETXCF START, REBUILD
- **M**: The rebuild caused by the maximum number of lock structure users being reached.
- **A**: The dynamic expansion or contraction initiated by the MVS operator command SETXCF START, ALTER
- **D**: Rebuild started to establish DUPLEX
- **P**: Duplexing being stopped, falling back to primary.
- **W**: Duplexing being stopped, switching to secondary.
- **S**: Dynamic expand/contract initiated by MVS SETXCF START,ALTER operator command against a secondary group buffer pool.

**Field Name**: QW0267RS

**STRUCTURE NAME**

The name of the CF structure.

**Field Name**: QW0267NM

**REQUESTED SIZE**

The requested size of the CF structure in 4 KB increments. This field is valid only if the value in the REASON is ALTER COMMAND.

**Field Name**: QW0267SZ
IFCID 268 - CF Rebuild/Alter End

This topic shows detailed information about “Record Trace - IFCID 268 - CF Rebuild/Alter End”.

This record shows the end of a coupling facility (CF) alter or rebuild. This end record matches the start record, IFCID 267.

Record trace - IFCID 268 - CF Rebuild/Alter End

The field labels shown in the following sample layout of “Record Trace - IFCID 268 - CF Rebuild/Alter End” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPERATION</td>
<td>The operation for the DB2 data sharing coupling facility structures:</td>
</tr>
<tr>
<td>F</td>
<td>The rebuild due to the coupling facility structure failure or loss of connectivity to the coupling facility.</td>
</tr>
<tr>
<td>O</td>
<td>The MVS rebuild initiated by the MVS operator command SETXCF START, REBUILD</td>
</tr>
<tr>
<td>M</td>
<td>The rebuild caused by the maximum number of lock structure users being reached.</td>
</tr>
<tr>
<td>A</td>
<td>The dynamic expansion or contraction by the MVS operator command SETXCF START, REBUILD</td>
</tr>
<tr>
<td>D</td>
<td>Rebuild started to establish DUPLEX</td>
</tr>
<tr>
<td>P</td>
<td>Duplexing being stopped, falling back to primary.</td>
</tr>
<tr>
<td>W</td>
<td>Duplexing being stopped, switching to secondary.</td>
</tr>
<tr>
<td>S</td>
<td>Dynamic expand/contract initiated by MVS SETXCF START,ALTER operator command against a secondary group buffer pool.</td>
</tr>
</tbody>
</table>

Field Name: QW0268FC

START TIME

The date and time of the start of the rebuild.

Field Name: QW0268BT

OPERATION RESULT

The result of the operation:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>O</td>
<td>The operation completed successfully.</td>
</tr>
<tr>
<td>N</td>
<td>The expansion or contraction completed successfully, however, the allocated size is smaller than the requested size.</td>
</tr>
<tr>
<td>S</td>
<td>The rebuild, expansion, or contraction was stopped.</td>
</tr>
</tbody>
</table>

FIELD NAME: QW0268RC

REASON STOPPED

40-758
The reason why the rebuild, expansion, or contraction was stopped:

C  Duplexing rebuild stopped because of insufficient connectivity due to a change in the set of connectors
F  Structure failed before the operation completed
G  An MVS service failed before the operation completed
I  New structure does not provide connectivity which is better than or equivalent to the current structure
J  The structure alter request could not complete due to a rebuild initiated for the structure
K  Rebuild process was stopped because of failure on connect to the new structure
L  Lost connectivity to the structure
N  New structure does not provide better connectivity than the current structure for a LossConn rebuild
O  Operator requested to stop
P  Duplexing was stopped by new CFRM policy
R  Resource manager requested to stop
S  Invalid ratio specified
T  Rebuild process was stopped because the new lock structure is full
U  Rebuild process was stopped because of failure of a required IRLM in the group
W  Rebuild stopped due to successful group function level change--complete rebuild is not required
X  Rebuild stopped due to unsuccessful completion of group function level change

This field is only valid if the value in OPERATION RESULT is S.

Field Name: QW0268RS

DIRECTORY COUNT

If the structure was altered, this is the current directory count of the directory entries. If the GBP was rebuilt, this field is not used. For the SCA and lock structure, this is a serviceability field.

Field Name: QW0268DN

STRUCTURE NAME

The name of the CF structure.

Field Name: QW0268NM

ELEMENT COUNT

If the structure was altered, this is the current count of the elements. For an GBP with a 8 KB page size, the element count equals the data entry count. For a GBP with a 16 KB page size, the element count is eight times the data entry count. If the GBP was rebuilt, this field is not used. For the SCA and lock structure, this is a serviceability field.

Field Name: QW0268TN
MINIMUM SIZE

If the structure was altered, this is the current minimum structure in increments of 4 KB. If the GBP was rebuilt, this is number of pages cast out by this member.

Field Name: QW0268MS

CURRENT SIZE

If the structure was altered, this is the current structure size in increments of 4 KB. If the GBP was rebuild, this is number of pages written to the new structure by this member.

Field Name: QW0268CS
### IFCID 269 - Trusted/Context Trace

This topic shows detailed information about “Record Trace - IFCID 269 - Trusted/Context Trace”.

This record is produced, if a trusted connection is established or reused.

**Record trace - IFCID 269 - Trusted/Context Trace**

The field labels shown in the following sample layout of “Record Trace - IFCID 269 - Trusted/Context Trace” are described in the following section.

<table>
<thead>
<tr>
<th>CONNECTION TYPE</th>
<th>STATUS: FAILED</th>
<th>SQLCODE: 20360</th>
<th>OBJECT OWNER: N/P</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRUSTED CONTEXT NAME</td>
<td>CON1</td>
<td>SECURITY LABEL : N/P</td>
<td></td>
</tr>
<tr>
<td>SYSTEM AUTHID USED</td>
<td>KOZS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ROLE ASSOCIATED</td>
<td>MYROLE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TCP/IP ADDRESS</td>
<td>ADDR1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SERVAUTH NAME</td>
<td>XXXXXXXXXXXXXXXXXXXXXXZ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ENCRYPTION</td>
<td>CCCCCCCCCCCCCC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JOB NAME</td>
<td>DDDDDDDDD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>REUSE AUTHID</td>
<td>EEEEEEEEE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>USER ROLE</td>
<td>my USER ROLE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PROFILE NAME</td>
<td>MYPROFILE</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**CONNECTION TYPE**

The type of trusted connection. Possible values are:

- **ESTABLISHED** or **ESTABLISH TRUSTED CONNECTION**
  
  If a trusted connection is established.

- **REUSED** or **REUSE TRUSTED CONNECTION**
  
  If a trusted connection is reused.

**Field Name:** QW0269TY

**STATUS**

The status of the trusted connection:

- **SUCCESS**
  
  If a trusted connection was established or reused successfully.

- **FAILED** or **FAILURE**
  
  If a trusted connection failed, when it was tried to be established or reused.

If the status is neither **SUCCESS** nor **FAILURE**, the value itself is shown.

**Field Name:** QW0269ST

**SQLCODE**

The SQL code returned after running the SQL statement.

**Field Name:** QW0269SQ

**OBJECT OWNER**

The owner of the objects that are created using the trusted context:

- **ROLE**
  
  The role.
IFCID 269 - Trusted/Context Trace

AUTHID
The AUTHORIZATION ID.
Field Name: QW0269OT

SECURITY LABEL
The security label.
Field Name: QW0269SL

TRUSTED CONTEXT NAME
The trusted context name.
Field Name: QW0269TC

SYSTEM AUTHID USED
The system authorization ID that is used to establish the trusted connection.
Field Name: QW0269SA

ROLE ASSOCIATED
The default role associated with the context.
Field Name: QW0269RC

TCP/IP ADDRESS
The actual communication TCP/IP address used for connection.
Field Name: QW0269AD

SERVAUTH NAME
The SERVAUTH name of the TCP/IP security zone.
Field Name: QW0269SR

ENCRYPTION
The encryption value to be associated with the encryption trust attribute for a trusted context. Possible values are:
- NONE
- LOW
- HIGH
Field Name: QW0269EC

JOB NAME
The job name for a local application.
Field Name: QW0269JN

REUSE AUTHID
The authorization ID under which a trusted connection is reused.
Field Name: QW0269RA

USER ROLE
The user role.
Field Name: QW0269RU

PROFILE NAME
IFCID 269 - Trusted/Context Trace

The RACF profile name that contains the authorization IDs that can use the connection in the trusted context.

Field Name: QW0269PR
IFCID 270 - Trusted/Context Trace

This topic shows detailed information about “Record Trace - IFCID 270 - Trusted/Context Trace”.

This record is produced, if a trusted connection is created or altered.

Record trace - IFCID 270 - Trusted/Context Trace

The field labels shown in the following sample layout of “Record Trace - IFCID 270 - Trusted/Context Trace” are described in the following section.

STATEMENT TYPE: CREATE  SQLCODE: 11  SQL STMT LENGTH: 12
SQL STATEMENT : CREATE MYTAB

STATEMENT TYPE
The type of trusted context. Possible values are:

CREATE TRUSTED CONTEXT or CREATE
If a trusted context is created.

ALTER TRUSTED CONTEXT or ALTER
If a trusted context is altered.

Field Name: QW0270TY

SQLCODE
The SQL return code from the CREATE or ALTER TRUSTED CONTEXT statement.

Field Name: QW0270SQ

SQL STMT LENGTH
The length of the SQL statement.

Field Name: QW0270SL

SQL STATEMENT
The SQL statement (truncated at 4000 bytes).

Field Name: QW0270SS
IFCID 271 - Row Level and Column Level Access Control

This topic shows detailed information about “Record Trace - IFCID 271 - Row Level and Column Level Access Control”.

This IFCID records the following events:
- When a row permission or column mask is created.
- When a row permission or column mask is dropped.
- When a row permission or column mask is altered.

Record trace - IFCID 271 - Row Level and Column Level Access Control

The field labels shown in the following sample layout of “Record Trace - IFCID 271 - Row Level and Column Level Access Control” are described in the following section.

<table>
<thead>
<tr>
<th>STATEMENT TYPE: CREATE</th>
<th>OBJECT: ROW PERMISSION</th>
<th>SQLCODE</th>
<th>STMT LENGTH</th>
<th>SQL STATEMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>CREATE or C</td>
<td></td>
<td></td>
<td>78</td>
<td>THIS IS A SQL STATEMENT TEXT TO VERIFY ITS CORRECT PRESENTATION IN BATCH AUDIT</td>
</tr>
</tbody>
</table>

**TYPE**

Identifies the SQL statement type:
- CREATE or C
  - Creates row permission or column mask.
- DROP or D
  - Drops row permission or column mask.
- ALTER or A
  - Alters row permission or column mask.

Otherwise, a hexadecimal value is shown.

**Field Name:** QW0271TY

**OBJECT**

Identifies the object type:
- Row permission (R)
- Column mask (M)

Otherwise, a hexadecimal value is shown.

**Field Name:** QW0271OB

**SQLCODE**

The SQL code from the execution of the CREATE, DROP, or ALTER statement.

**Field Name:** QW0271SQ

**STMT LENGTH**

The length of the SQL statement.

**Field Name:** QW0271SL

**SQL STATEMENT**

The SQL statement text associated with the table access. The maximum length is 4000 bytes. Long SQL text can be truncated.
IFCID 271 - Row Level and Column Level Access Control

Field Name: QW0271SS
IFCID 272 - Associate Locators

This topic shows detailed information about “Record Trace - IFCID 272 - Associate Locators”.

Record trace - IFCID 272 - Associate Locators

The field labels shown in the following sample layout of “Record Trace - IFCID 272 - Associate Locators” are described in the following section.

LOCATION
The location name where the stored procedure executes.
Field Name: QW0272LN

PKG COLLECTION ID
The Package collection identifier. This is BLANK when the statement executes without a package.
Field Name: QW0272PC

PROGRAM NAME
The program name.
Field Name: QW0272PG

STO PROC LOCATION
The location of the stored procedure.
Field Name: QW0272LP

STO PROC QUALIFIER
The qualifier of the stored procedure.
Field Name: QW0272QN

STO PROC NAME
The name of the stored procedure.
Field Name: QW0272PN

STATEMENT NUMBER
The statement number of ASSOCIATE LOCATORS statement.
Field Name: QW0272SN

NUMBER OF LOCATORS
The number of locators referenced in the ASSOCIATE LOCATORS statement.
Field Name: QW0272NL
IFCID 272 - Associate Locators

CONSISTENCY TOKEN

The consistency token.

Field Name: QW0272TS
IFCID 273 - Allocate Cursor

This topic shows detailed information about “Record Trace - IFCID 273 - Allocate Cursor”.

Record trace - IFCID 273 - Allocate Cursor

The field labels shown in the following sample layout of “Record Trace - IFCID 273 - Allocate Cursor” are described in the following section.

**LOCATION NAME**

The location name where the store procedure executes.

Field Name: QW0273LN

**PKG COLLECTION ID**

The Package collection identificator. This is BLANK when the statement executes without a package.

Field Name: QW0273PC

**PROGRAM NAME**

The program name.

Field Name: QW0273PG

**STO PROC LOCATION**

The location of the stored procedure.

Field Name: QW0273LP

**STO PROC QUALIFIER**

The qualifier of the stored procedure.

Field Name: QW0273QN

**STO PROC NAME**

The name of the stored procedure.

Field Name: QW0273PN

**ALLOCATED CURSOR**

The name of the ALLOCATE CURSOR statement.

Field Name: QW0273CN

**REAL CURSOR NAME**

The name of cursor in the stored procedure.
IFCID 273 - Allocate Cursor

Field Name: QW0273RN
STATEMENT NUMBER
The statement number of ALLOCATE CURSOR statement.
From QW0273SN or QW0273TS.
Field Name: RT0325SN
LOCATOR VALUE
The value of the locator associated with the result set for which this cursor is defined.
Field Name: QW0273LV
CONSISTENCY TOKEN
The consistency token.
Field Name: QW0273TS
QUERY COMMAND ID
The ID of the query command.
Field Name: QW0273CID
QUERY INSTANCE ID
The ID of the query instance.
Field Name: QW0273QID
IFCID 305 - Table Check Constraint

This topic shows detailed information about “Record Trace - IFCID 305 - Table Check Constraint”.

Record trace - IFCID 305 - Table Check Constraint

The field labels shown in the following sample layout of “Record Trace - IFCID 305 - Table Check Constraint” are described in the following section.

NAME
The check constraint name.
Field Name: QW0305CN

TEXT
The check constraint text.
Field Name: QW0305CT

DBID
The DBID of the database for the table on which the check constraint is defined.
Field Name: QW0305DB

OBID
The OBID of the table on which the check constraint is defined.
Field Name: QW0305OB

OPERATION
The operation that is utilizing the check constraint function:
CREATE A check constraint is defined with a CREATE TABLE operation.
ALTER ADD A check constraint is defined with an ALTER TABLE operation.
ALTER ADD ENFORCE A check constraint is enforced during an ALTER TABLE operation.
ALTER ADD DROP A check constraint is removed with an ALTER TABLE operation.
ENFORCE A check constraint is enforced. DB2 checks that a row does not violate a check constraint.
Field Name: QW0305OP

RESULT
The result of the enforced check constraint:
IFCID 305 - Table Check Constraint

<table>
<thead>
<tr>
<th>Key</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>REJ</td>
<td>The check constraint was rejected due to a check constraint violation.</td>
</tr>
<tr>
<td>OK</td>
<td>No check constraint was violated.</td>
</tr>
</tbody>
</table>

Field Name: QW0305RS

RECORD IDENTIFIER

The record identifier (RID) of the record that failed the check constraint. This field is only valid if the value in RESULT is REJ.

Field Name: QW0305ID

TABLE_SPACE_TYPE

The type of the table space:

- L  Non-EA large table
- N  Non-large table
- V  EA-enabled large table

Field Name: QW0305TY

CHARACTERS

The first 30 characters of the rejected record that failed the check constraint condition. This field is only valid if the value in RESULT is REJ. Otherwise, N/A is printed in this field.

Field Name: QW0305RR
IFCID 311 - Global Temp Table Usage

This topic shows detailed information about “Record Trace - IFCID 311 - Global Temp Table Usage”.

Record trace - IFCID 311 - Global Temp Table Usage

The field labels shown in the following sample layout of “Record Trace - IFCID 311 - Global Temp Table Usage” are described in the following section.

**TEMP TAB CREATOR**

The creator of the global temporary table.

Field Name: QW0311QN

**TEMP TABLE NAME**

The name of the global temporary table.

Field Name: QW0311TN

**PACK LOCATION NAME**

The package location name for the query that uses the global temporary table.

Field Name: QW0311LN

**PROGRAM NAME**

The program name for the query that uses the global temporary table.

Field Name: QW0311PN

**CURSOR NAME**

The cursor name for fetches. This field is only applicable if the value in WORKFILE TYPE is RC. Otherwise, N/A is printed.

Field Name: QW0311CN

**PACK COLLECTION ID**

The package collection identifier for the query that uses the global temporary table.

Field Name: QW0311PC

**PACK VERSION**

The package version for the query that uses the global temporary table.

Field Name: QW311PVF

**CURSOR HOLD STATUS**

The cursor hold status:

- **HO** The cursor is held through commit.
- **'BLANK'** The cursor is not held through commit.
IFCID 311 - Global Temp Table Usage

Field Name: QW0311HO

WORKFILE TYPE

The work-file type:
TT Temporary table
C Cursor on a temporary table
TR Transition table
CT Cursor on transition table.

Field Name: QW0311TY

OPERATION

The operation using the global temporary table:
AT Alter the temporary table.
CI Create the temporary table instantiation. A work file is created for the temporary table.
OC Open the cursor on a temporary table.
D Delete work files for temporary table.
DA Delete all rows from the temporary table, but leave the work-file structures intact.
CC Close cursor on the temporary table.

Field Name: QW0311OP

QW0311CA

This field is for IBM service use.

Field Name: QW0311CA

QW0311TA

This field is for IBM service use.

Field Name: QW0311TA
IFCID 313 - Uncommitted Unit of Recovery

This topic shows detailed information about “Record Trace - IFCID 313 - Uncommitted Unit of Recovery”.

Uncommitted unit of recovery (UR) shows data from IFCID 313. It reflects the same information given in the DB2 messages DSNR035I and DSNR036I.

Record trace - IFCID 313 - Uncommitted Unit of Recovery

The field labels shown in the following sample layout of “Record Trace - IFCID 313 - Uncommitted Unit of Recovery” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UNCOMMITTED URID</td>
<td>The ID of the uncommitted unit of recovery. Field Name: QW0313ID</td>
</tr>
<tr>
<td>CHKPTS TAKEN</td>
<td>For inflight units of recovery (UR), the number of checkpoints taken since the beginning of the UR. For indoubt URs, this field is set to -1. Field Name: QW0313CK</td>
</tr>
<tr>
<td>TYPE OF UR/UW</td>
<td>The type of uncommitted unit of recovery (UR/UW): Field Name: QW0313TY</td>
</tr>
<tr>
<td>LUWID - NETWORKID, LUNAME, INSTANCE, COMMIT COUNT</td>
<td>The logical unit of work ID (LUWID) identifies the thread within the network. It consists of the: Field Name: QW0313LU</td>
</tr>
<tr>
<td>CONNECTION ID</td>
<td>The connection ID. Field Name: QW0313CN</td>
</tr>
<tr>
<td>CORRELATION ID</td>
<td></td>
</tr>
</tbody>
</table>
IFCID 313 - Uncommitted Unit of Recovery

The correlation ID.

Field Name: QW0313CR

MESSAGE NUMBER

The number of the DB2 message reflecting the information in this IFCID.

Field Name: QW0313MG

PLAN NAME

Plan Name from URE, if the UR is active, from RURE, if the UR is inactive (indoubt).

Field Name: QW0313PN

LOG RECS WRTN

Shows one of the following:
- The number of log records written
- The total number of minutes that the reader has been running

Field Name: QW0313LW

THRESHOLD TYPE

The type of threshold reached:
- C  Checkpoints
- L  Log records

Field Name: QW0313TH

AUTHORIZATION ID

Authorization ID from URE, if the UR is active, from RURE, if the UR is inactive (indoubt).

Field Name: QW0313AI

END USER USERID

End-user ID from CCB, if the UR is active. End user information is not available for indoubt URs.

Field Name: QW0313EU

TRANSACTION

End-user transaction name from CCB, if the UR is active. End user information is not available for indoubt URs.

Field Name: QW0313ET

WORKSTATION

End-user workstation name from CCB if the UR is active. End user information is not available for indoubt URs.

Field Name: QW0313EW
IFCID 314 - Authorization Exit Parameters

This topic shows detailed information about “Record Trace - IFCID 314 - Authorization Exit Parameters”.

It is generated after the authorization exit is called and shows the contents of the parameter list.

This record can be useful when debugging an authorization exit.

Record trace - IFCID 314 - Authorization Exit Parameters

The field labels shown in the following sample layout of “Record Trace - IFCID 314 - Authorization Exit Parameters” are described in the following section.

ADDRESS EXPL : X'000000AA'
ADDRESS WORK AREA: X'000000BB'
AUTH ID : N/P
UNQUALIFIED OBJECT NAME: N/P
OBJECT OWNER : N/P
RELATED INFO 1 : N/P
RELATED INFO 2 : N/P
LENGTH WORK AREA : 204
ACEE UTOKEN : UTOKEN01XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
PARAMETER LIST :
0000 C1C2C3C4 C5C6C7C8 C9404040 40404040 40404040 40404040 40404040 40404040 | ABCDEFGHI
0020 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 |
0040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 |
0060 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 |
0080 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 |
00A0 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 |
00C0 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 |
00E0 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 |

ADDRESS EXPL

The address of the exit parameter list.
Field Name: QW0314EL

EXIT RETURN CODE

The return code from the exit:
0 Access allowed.
4 Check the DB2 authorization.
8 Access denied.
12 Unable to determine authorization. Do not call the exit again.

Field Name: QW0314RC

STO CLOCK BEFORE EXIT CALL

The store clock value before the exit was called.
Field Name: QW0314BC

ADDRESS WORK AREA

The address of the work area.
Field Name: QW0314WA

EXIT REASON CODE

The reason code from the user-defined exit.
Field Name: QW0314RS

STO CLOCK AFTER EXIT CALL

Field Name: QW0314RS
IFCID 314 - Authorization Exit Parameters

The store clock value after the exit was called.
Field Name: QW0314AC

AUTH ID
The authorization ID that is checked by DB2.
Field Name: QW0314UN

UNQUALIFIED OBJECT NAME
The unqualified object name.
Field Name: QW0314BN

OBJECT OWNER
The object owner or qualifier.
Field Name: QW0314ON

RELATED INFO 1
Shows other related information in field 1.
Field Name: QW03141N

RELATED INFO 2
Shows other related information in field 2.
Field Name: QW03142N

LENGTH WORK AREA
The length of the work area.
Field Name: QW0314WL

ACEE UTOKEN
Shows the ACEE UTOKEN, if it is available. If it is not available, the first word of this field contains one of the following values:

0  The UTOKEN cannot be accessed
-1  An abend occurred during the attempt to access the ACEE.

Field Name: QW0314UT

PARAMETER LIST
The list of parameters specific to the exit.
Field Name: QW0314PL
IFCID 316 - SQL Statement Statistics

This topic shows detailed information about “Record Trace - IFCID 316 - SQL Statement Statistics”.

IFCID 316 reports on the contents of the prepared SQL statement cache. This record is only written when an IFI application requests IFCID 316 through the READS interface.

It provides one record for each qualifying SQL statement in the cache. These multiple records are placed in the output area provided by the IFI application. The IFI application can specify qualification criteria for which statements should be reported.

Record trace - IFCID 316 - SQL Statement Statistics

The field labels shown in the following sample layout of “Record Trace - IFCID 316 - SQL Statement Statistics” are described in the following section.

STATEMENT NAME : X'00650B786B4536444B3A2A490000003C' STATEMENT IDENTIFIER : 593
LITERAL REPLACEMENT : NO LINE NUMBER : 0
STATUS : INVALIDATED BY DROP OR ALTER

TIME STATISTICS COLLECTION START: 12/20/10 16:34:52.484040
TIME STATEMENT STORED IN CACHE : 12/22/10 11:36:42.874097 IN STORE CLOCK FORMAT : X'C710D9CFD18F1F1A'
TIME STATEMENT UPDATED IN CACHE : 12/22/10 11:36:42.874778 IN STORE CLOCK FORMAT : X'20101222113642874778'

STATEMENT COPIES : 0 STATEMENT EXECUTIONS : 1
SYNCH BUFFER READS : 0 SYNCH BUFFER WRITES : 0
CURRENT USERS : 0 GETPAGE OPERATIONS : 2
TABLESPACE SCANS : 0 PARALLEL GROUPS CREATED : 0
ROWS EXAMINED : 0 ROWS PROCESSED : 1
SORTS : 0 INDEX SCANS : 0
ACCUMULATED CPU TIME : 0.000210 ACCUMULATED ELAPSED TIME : 0.000218

RID LIST SECTION

(HJA=HYBRID JOIN APPEND, IA=INDEX ACCESS, OV=OVERFLOW, RL=RID LIST)
RL NOT USED LIMIT EXCEEDED: 0 RL NOT USED NO STOR AVAIL : 0
RL OV - NO POOL STOR AVAIL: 0 RL OV - RIDS EXCEED LIMIT : 0
HJA - NO POOL STOR AVAIL : 0 HJA - RIDS EXCEED LIMIT : 0
RL RETRIEVAL IA SKIPPED : 0

READ BY OTHER THREAD : N/P WRITE BY OTHER THREAD : N/P
GLOBAL LOCKS : N/P LATCH REQUEST : N/P
DRAIN LOCK : N/P LOG WRITER : N/P

CURRENT DATA BIND OPTION : NO CURSOR WITH HOLD : NO
CURRENT PRECISION SPEC REG: DEC15 DYNAMIC RULES BIND OPTION : RUN
CURRENT DEGREE SPECIAL REG: 1 ISOLATION BIND OPTION : CURSOR STABILITY
CURRENT RULES SPECIAL REG : DB2 DSG MEMBER : 'BLANK'

TRANSACTION NAME : db2jcc_application
SIGNON USER ID : skadm WORKSTATION ID : candlelight
PROGRAM NAME : SYSLH200 USER ID : skadm
USER GROUP : skadm OBJECT QUALIFIER : skadm
REFERENCED TABLE QUALIFIER : skadm REFERENCED TABLE NAME : LE105
USER PROVIDED ID STRING : N/P CURRENT SCHEMA : skadm

LENGTH OF SQL STATEMENT : 60 SQL STATEMENT - FIRST 60 BYTES : INSERT INTO LE105 (ID, NAME, SALARY) VALUES(1, 'dummy', 100)

STATEMENT NAME

The name of the statement generated by DB2.

Field Name: QW0316NM

STATEMENT IDENTIFIER
IFCID 316 - SQL Statement Statistics

The unique identifier of the statement. A number is generated to uniquely identify a statement in the prepared statement cache.

Field Name: QW0316TK

LITERAL REPLACEMENT

Indicates the cache literal replacement. Possible values are:

NO   No literal replacement was done.
REPLACE   Literals were replaced in the statement.
DUPLICATE   Literals were replaced in the statement, but the cached statement is a duplicate of another statement in the cache. A match with the other statement in the cache failed only because the literal reuse criteria were not met.

Field Name: QW0316LR

LINE NUMBER

The precompiler line number of the initial PREPARE statement.

Field Name: QW0316LX

STATUS

The status of the statement. If any of the following flags are set, the statement has actually been removed from the cache but current users might still have an active copy. DB2 will continue to track the statement until the use-count and copy-count are zero.

It can be one of the following:

• INVALIDATED BY DROP OR ALTER
• INVALIDATED BY REVOKE
• REMOVED FROM CACHE BY LRU
• INVALIDATED BY UTILITIES

Field Name: QW0316FL

TIME STATISTICS COLLECTION START

Shows the time stamp when the statistics collection began.

Field Name: QW0316TS

TIME STATEMENT STORED IN CACHE

The date and time when the statement was inserted into the cache (in DB2 timestamp format).

Field Name: QW0316TM

IN STORE CLOCK FORMAT

The date or time when the statement was inserted into the cache (in store clock format) (DB2 field: QW0316TM2).

Field Name: RT316TM2

TIME STATEMENT UPDATED IN CACHE

The date and time when the statement was updated, in internal format (DB2 field: QW0316UT2).
Field Name: RT316UT2

IN STORE CLOCK FORMAT

The date or time when the statement statistic was updated (in store clock format) (DB2 field: QW0316UT1).

Field Name: RT316UT1

STATEMENT COPIES

The number of copies of the statement owned by all threads in the system.

Note: This includes QW0316US and any copies owned by plans or packages bound with KEEPDYNAMIC(YES) that were not used in their current unit of work. These users prepared the statement in a previous unit of work and still have it in a prepared state.

Field Name: QW0316CP

STATEMENT EXECUTIONS

The number of statement executions.

Note: For a cursor statement, this is the number of OPENS.

Field Name: QW0316NE

SYNCH BUFFER READS

The number of synchronous buffer read operations performed for the statement.

Field Name: QW0316NB

SYNCH BUFFER WRITES

The number of synchronous buffer write-operations performed for statement.

Field Name: QW0316NW

CURRENT USERS

Number of current users of the SQL statement.

Note: These users have prepared or executed the statement during their current unit of work.

Field Name: QW0316US

GETPAGE OPERATIONS

The number of Getpage operations performed for a statement.

Field Name: QW0316NG

TABLESPACE SCANS

The number of scan operations for table spaces that are performed for a statement.

Field Name: QW0316NT

PARALLEL GROUPS CREATED

The number of parallel groups that are created for a statement.

Field Name: QW0316NL
IFCID 316 - SQL Statement Statistics

ROWS EXAMINED
The number of rows that are examined for the statement.
Field Name: QW0316NR

ROWS PROCESSED
The number of rows that are processed for the statement. For example, the number of rows returned for a SELECT statement, or the number of rows affected by an INSERT, UPDATE, or DELETE statement.
Field Name: QW0316NP

SORTS
The number of sort operations performed for a statement.
Field Name: QW0316NS

INDEX SCANS
The number of index scans performed for a statement.
Field Name: QW0316NI

ACCUMULATED CPU TIME
The accumulated CPU time.
Field Name: QW0316CT

ACCUMULATED ELAPSED TIME
Shows the accumulated elapsed time used for a statement.
Field Name: QW0316AE

RL NOT USED LIMIT EXCEEDED
The number of times that a RID list was not used, because the number of:
- RIDs would have exceeded one or more internal DB2 limits
- RID blocks exceeded the value set by the MAXTEMPS_RID system parameter.
Field Name: QW0316RT

RL NOT USED NO STOR AVAIL
The number of times that a RID list was not used because there was not enough storage. This also applies if the work file storage was not available.
Field Name: QW0316RS

RL OV - NO POOL STOR AVAIL
The number of times a RID list was overflowed to a work file because no RID pool storage was available to hold the list of RIDs (DB2 field: QW0316WFRIDS).
Field Name: RT316IDS

RL OV - RIDS EXCEED LIMIT
The number of times a RID list was overflowed to a work file because the number of RIDs exceeded one or more internal limits (DB2 field: QW0316WFRIDT).
Field Name: RT316IDT
HJA - NO POOL STOR AVAI

The number of times a RID list append for a hybrid join was interrupted because no RID pool storage was available to hold the list of RIDs (DB2 field: QW0316HJINCS). For example, the number of times DB2 interrupted the RID phase and switched to the data phase.

Field Name: RT316NCS

HJA - RIDS EXCEED LIMIT

The number of times a RID list append for a hybrid join was interrupted because the number of RIDs exceeded one or more internal limits (DB2 field: QW0316HJINCT). For example, it shows the number of times DB2 interrupted the RID phase and switched to the data phase.

Field Name: RT316NCT

RL RETRIEVAL IA SKIPPED

The number of times a RID list retrieval for multiple index access was skipped because DB2 predetermined the outcome of index ANDing or ORing (DB2 field: QW0316RSMIAPI).

Field Name: RT316IAP

READ BY OTHER THREAD

The accumulated wait time for a read activity that is performed by another thread.

Field Name: QW0316W5

SYNCH EXECUTION UNIT SWITCH

The accumulated wait time for a synchronous execution unit switch.

Field Name: QW0316W3

WRITE BY OTHER THREAD

The accumulated wait time for a write activity that is performed by another thread.

Field Name: QW0316W6

SYNCHRONOUS I/O

The accumulated wait time for synchronous I/O.

Note: This wait time and the following wait times are only collected if a CLASS 3 accounting trace is started.

Field Name: QW0316W1

GLOBAL LOCKS

The accumulated wait time for global locks.

Field Name: QW0316W4

LOCK AND LATCH REQ

The accumulated wait time for lock and latch requests.

Field Name: QW0316W2

LATCH REQUEST

The accumulated wait time for lock requests.
Field Name: QW0316W7

PAGE LATCH
The accumulated wait time for page latches.
Field Name: QW0316W8

DRAIN LOCK
The accumulated wait time for drain locks.
Field Name: QW0316W9

DRAIN WAITING FOR CLAIM REL
The accumulated wait time for drains when waiting for claims to be released.
Field Name: QW0316WA

LOG WRITER
The accumulated wait time for log writers.
Field Name: QW0316WB

CURRENT DATA BIND OPTION
The CURRENTDATA bind option. It can be one of the following:
• YES
• NO
Field Name: QW0316X7

CURSOR WITH HOLD
Shows if the position for a cursor opened WITH HOLD. It can be one of the following:
• YES
• NO
Field Name: QW0316XC

CURRENT PRECISION SPEC REG
Shows the CURRENT PRECISION special register. It can be one of the following:
• DEC31
• DEC15
Field Name: QW0316XB

DYNAMIC RULES BIND OPTION
Shows the dynamic rules bind option. It can be one of the following:
• BIND
• RUN
Field Name: QW0316X8

CURRENT DEGREE SPECIAL REG
Shows value of the CURRENT DEGREE special register. It can be one of the following:
• ANY
ISOLATION BIND OPTION

The value of the ISOLATION bind option that is in effect for the initial PREPARE statement. It can be one of the following:
  • UNCOMMITTED READ
  • CURSOR STABILITY
  • READ STABILITY
  • REPEATABLE READ

Note: This value does not reflect if it is specified in a WITH clause.

CURRENT RULES SPECIAL REG

Shows the value of the CURRENT RULES special register. It can be one of the following:
  • DB2
  • SQL

TRANSACTION NAME

The text of the transaction name.

SIGNON USER ID

The End User ID is provided during RRS signon or resignon for initial prepare.

WORKSTATION ID

The Workstation ID is provided during RRS signon or resignon for initial prepare.

PROGRAM NAME

The text of the program name.
IFCID 316 - SQL Statement Statistics

The name of the user group. The user group is the current SQLID of the user who started the initial PREPARE statement.

Field Name: QW0316X4

OBJECT QUALIFIER
The qualifier that is used for unqualified table names.

Field Name: QW0316X5

REFERENCED TABLE QUALIFIER
The qualifier of the referenced table name.

Field Name: QW0316QD

REFERENCED TABLE NAME
The name of the referenced table.

Field Name: QW0316TD

USER PROVIDED ID STRING
The identification (ID) string provided by the user.

Field Name: QW0316UI

CURRENT SCHEMA
The special register text of the current schema.

Field Name: QW0316SC

LENGTH OF SQL STATEMENT
The length of the entire statement.

Field Name: QW0316LN

SQL STATEMENT - FIRST 60 BYTES
The first 60 bytes of the SQL statement text.

Field Name: QW0316TX
IFCID 317 - SQL Statement String

This topic shows detailed information about “Record Trace - IFCID 317 - SQL Statement String”.

Record trace - IFCID 317 - SQL Statement String

The field labels shown in the following sample layout of “Record Trace - IFCID 317 - SQL Statement String” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SQL STATEMENT NAME</td>
<td>The name of the SQL statement.</td>
</tr>
<tr>
<td>SQL STATEMENT IDENTIFIER</td>
<td>The identifier of the SQL statement in hexadecimal.</td>
</tr>
<tr>
<td>SQL STATEMENT LENGTH</td>
<td>The length of the SQL statement.</td>
</tr>
<tr>
<td>SQL STATEMENT TEXT</td>
<td>The text of the SQL statement.</td>
</tr>
<tr>
<td>ATTRIBUTE STRING LENGTH</td>
<td>The length of the attribute string.</td>
</tr>
<tr>
<td>ATTRIBUTE STRING TEXT</td>
<td>The text of the attribute string.</td>
</tr>
</tbody>
</table>

SQL STATEMENT NAME

Field Name: QW0317NM

SQL STATEMENT IDENTIFIER

Field Name: QW0317ID

SQL STATEMENT LENGTH

Field Name: QW0317LN

SQL STATEMENT TEXT

Field Name: QW0317TX

ATTRIBUTE STRING LENGTH

Field Name: QW03172LN

ATTRIBUTE STRING TEXT

Field Name: QW03172TX
IFCID 319 - Audit Security Record

This topic shows detailed information about “Record Trace - IFCID 319 - Audit Security Record”.

When a local DB2 receives a non-RACF identity that represents a user, it maps that name to a local user ID for use in connection processing. This record traces the mapping. This record provides an audit trail for security processing.

Record trace - IFCID 319 - Audit Security Record

The field labels shown in the following sample layout of “Record Trace - IFCID 319 - Audit Security Record” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>REQ COMMUNICATION ADDR</td>
<td>Requesting communication address. For SNA, this field shows the LU name, for TCP/IP, this shows the dotted decimal IP address.</td>
</tr>
<tr>
<td>COMMUNICATION ADDR TYPE</td>
<td>Type of communication address: SNA or TCP/IP.</td>
</tr>
<tr>
<td>CLIENT PRODUCT ID</td>
<td>The identification of the client product.</td>
</tr>
<tr>
<td>DERIVED LOCAL USERID</td>
<td>Local user ID mapped by DB2.</td>
</tr>
<tr>
<td>SECURITY TYPE</td>
<td>The type of security identity. Possible values are:</td>
</tr>
<tr>
<td>SECURITY MECHANISM</td>
<td>The security mechanism used. Possible values are:</td>
</tr>
</tbody>
</table>

**REQ COMMUNICATION ADDR**

Requesting communication address. For SNA, this field shows the LU name, for TCP/IP, this shows the dotted decimal IP address.

**Field Name**: QW0319AD

**COMMUNICATION ADDR TYPE**

Type of communication address: SNA or TCP/IP.

**Field Name**: QW0319CT

**CLIENT PRODUCT ID**

The identification of the client product.

**Field Name**: QW0319CP

**DERIVED LOCAL USERID**

Local user ID mapped by DB2.

**Field Name**: QW0319US

**SECURITY TYPE**

The type of security identity. Possible values are:

- KERBEROS
- ENCRYPTED

**Field Name**: QW0319TY

**SECURITY MECHANISM**

The security mechanism used. Possible values are:

- User ID (UID) and encrypted password (PW)
- Encrypted UID and PW
IFCID 319 - Audit Security Record

- Encrypted UID, PW, and new PW
- Encrypted UID and data
- Encrypted UID, PW, and data
- Encrypted UID, PW, new PW, and data
- Encrypted UID only

Field Name: QW0319SM

FLAGS - USER REGISTRY NAME

This flag shows if the caller passed the user registry name.

Field Name: QW0319UR

FLAGS - AES IS USED

This flag shows if Advanced Encryption Standard (AES) is used for encryption.

Field Name: QW0319AE

IPV6 ADDRESS

If the type of the communication address is TCP/IP, it is the 16 byte hexadecimal (HLHLHLHLHLHLHLHLHLHLHLHLHLHLHLHLHLHLHLHLHLHLHLHLHLHL) IP address of the internal 128 bit format, where:
- $H$ represents the high order half byte value
- $L$ represents the low order half byte value

Field Name: QW0319IPA

PRINCIPAL NAME LENGTH

Length of principal name.

Field Name: QW0319L1

PRINCIPAL NAME

The requesting principal name. This can be up to 256 characters and can contain lowercase characters.

Field Name: QW0319D1

PORT-INTERNAL FORMAT

If the type of the communication address is TCP/IP, this field shows the 16 bit port number in internal format.

Field Name: QW0319PRT
IFCID 321 - Force-at-Commit Begin

This topic shows detailed information about “Record Trace - IFCID 321 - Force-at-Commit Begin”.

Record trace - IFCID 321 - Force-at-Commit Begin

The field labels shown in the following sample layout of “Record Trace - IFCID 321 - Force-at-Commit Begin” are described in the following section.

TARGET LOCATION FOR WRITE X

TARGET LOCATION FOR WRITE
The target location for write.

Field Name: QW0321LO
IFCID 322 - Force-at-Commit End

This topic shows detailed information about “Record Trace - IFCID 322 - Force-at-Commit End”.

Record trace - IFCID 322 - Force-at-Commit End

The field labels shown in the following sample layout of “Record Trace - IFCID 322 - Force-at-Commit End” are described in the following section.

PAGES_WRITTEN : 22

PAGES_WRITTEN

The number of pages written.

Field Name: QW0322NP
IFCID 324 - Function Resolution

This topic shows detailed information about “Record Trace - IFCID 324 - Function Resolution”.

Record trace - IFCID 324 - Function Resolution

The field labels shown in the following sample layout of “Record Trace - IFCID 324 - Function Resolution” are described in the following section.

QUERYNO : 100 PLANNAME : PLAN0001 APPLNAME : APPLIC01
BIND_TIME : N/P VERSION : VERSION001XXXXXXXXX2XXXXXXXXX3XXXXXXXXX4XXXXXXXXX5XXXXXXXXX6XXXZ
CONSIS_TOKEN : X'C3D6D5E2E3D6D2F1'
COLLECTION_ID : COLLECTION01XXXXXXX2XXXXXXXXX3XXXXXXXXX4XXXXXXXXX5XXXXXXXXX6XXXXXXXXX7XXXXXXXXX8XXXXXXXXX9XXXXXXXXX0XXXXX
PROGNAME : PROGRAM001XXXXXXXXX2XXXXXXXXX3XXXXXXXXX4XXXXXXXXX5XXXXXXXXX6XXXXXXXXX7XXXXXXXXX8XXXXXXXXX9XXXXXXXXX0XXXXX
CURRENT_PATH : CURRENT PATH01XXXXX2XXXXXXXXX3XXXXXXXXX4XXXXXXXXX5XXXXXXXXX6XXXXXXXXX7XXXXXXXXX8XXXXXXXXX9XXXXXXXXX0XXXXX

QUERYNO
The query number.
Field Name: QW0324QN

PLANNAME
The plan name.
Field Name: QW0324PN

COLLECTION_ID
The collection ID.
Field Name: QW0324CI

APPLNAME
The name of the application.
Field Name: QW0324AL

PROGNAME
The program name.
Field Name: QW0324PG

CONSIS_TOKEN
The consistency token.
Field Name: QW0324CT

BIND_TIME
The time stamp of the bind time.
Field Name: QW0324TS

VERSION
The version ID.
Field Name: QW0324VN

CURRENT_PATH
The current path.
Field Name: QW0324CP
IFCID 324 - Function Resolution

FUNCT_SCHEMA
A short SQL identifier, either ordinary or delimited, following the concept of qualified names consistent with the ANSI/ISO SQL92 standard.
Field Name: QW0324FS

FUNCT_NAME
The name of a function without a qualifier.
Field Name: QW0324FN

SPECIFIC_NAME
Identifies the particular function. The specific name must identify a specific function name in the explicitly or implicitly specified schema.
Field Name: QW0324FI

FUNCT_TYPE
The classification of the function:
SU Scalar UDF
TU Table UDF
Field Name: QW0324FY

VIEW_CREATOR
The name of the view creator if the function is referenced in a view definition.
Field Name: QW0324CV

VIEW_NAME
The name of the view if the function is referenced in a view definition.
Field Name: QW0324NV

QUERY_BLOCKNO
A number that identifies the query block number being explained.
Field Name: QW0324QB

FUNCT_TEXT
Contains the text of the function reference, function name, and parameters. It can be up to 254 characters long.
Field Name: QW0324FT
IFCID 325 - Trigger Activation

This topic shows detailed information about “Record Trace - IFCID 325 - Trigger Activation”.

Record trace - IFCID 325 - Trigger Activation

The field labels shown in the following sample layout of “Record Trace - IFCID 325 - Trigger Activation” are described in the following section.

STATEMENT NO
The statement number of the SQL statement that activated the trigger.  
Field Name: QW0325SN

SQL STATEMENT
Triggering SQL statement. Possible values are: 
D DELETE  
I INSERT  
U UPDATE  
Field Name: QW0325SS

COLLECTION ID
The collection ID of the package containing the statement that activated the trigger.  
Field Name: QW0325CO

PROG NAME
Program or package containing the statement that activated the trigger.  
Field Name: QW0325PR

TRIGGER NAME
Trigger name.  
Field Name: QW0325NM

EXT.TRIGGER NAME
External trigger name.  
Field Name: QW0325TX

SCHEMA NAME
Schema name of the trigger.

Field Name: QW0325SC

**TIMESTAMP**

Trigger timestamp.

Field Name: QW0325TS

**ACTIVATION TIME**

Possible values are:

A  Trigger activation time is AFTER.
B  Trigger activation time is BEFORE.

Field Name: QW0325AC

**ENTRY/EXIT TYPE**

Possible values are:

E  Trigger is starting.
X  Trigger is ending.

Field Name: QW0325ET

**GRANULARITY**

Possible values are:

R  Trigger granularity is FOR EACH ROW.
S  Trigger granularity is FOR EACH STATEMENT.

Field Name: QW0325GR

**EVALUATION**

Triggered action condition evaluation. Possible values are:

T  Triggered action tested TRUE
F  Triggered action tested FALSE
N  No triggered action condition.

Field Name: QW0325CN

**NESTING LEVEL**

Nesting level of the trigger.

Field Name: QW0325NL

**SQLCA CONTENTS**

This section contains the SQLCA fields. It is only printed if the value in the ENTRY/EXIT TYPE field is RETURNED.

Field Name: QW0325SQ
IFCID 329 - IXL Suspensions

Record trace - IFCID 329 - IXL Suspensions

The field labels shown in the following sample layout of “Record Trace - IFCID 329 - IXL Suspensions” are described in the following section.

GBP NAME

Name of the group buffer pool.

Field Name: QW0329GB

REQUEST TYPE

The request type can be one of the following:

- READ-DIRINFO (IXLCACHE)
- READ-COCLASS (IXLCACHE)
- CASTOUT-DATA (IXLCACHE)
- DELETE-NAME (IXLCACHE)
- RESET-REFBIT (IXLCACHE)
- FORCE (IXLFORCE)
- CONNECT (IXLCONN)
- DISCONNECT (IXLDISC)
- PROCESS-REFLIST (IXLCACHE)
- READ-DATA (IXLCACHE)
- READ-STGSTATS (IXLCACHE)
- READ-COSTATS (IXLCACHE)
- UNLOCK-CASTOUT (IXLCACHE)
- SET-RECLVCTR (IXLCACHE)
- WRITE-DATA (IXLCACHE)
- X-INVALIDATE (IXLCACHE)
- REGISTER-PAGE-LIST (IXLCACHE)
- WRITE-DATA TO SECONDARY (IXLCACHE)
- DELETE-NAME-LIST TO SECONDARY (IXLCACHE)
- DELETE-NAME TO SECONDARY (IXLCACHE)
- READ-STGSTATS TO SECONDARY (IXLCACHE)

Field Name: QW0329RT

ASYNC. WAIT TIME

Asynchronous wait time in microseconds.

Field Name: QW0329ST
This topic shows detailed information about “Record Trace - IFCID 330 - Active Log Space Shortage”.

This record is written of each group buffer pool present. Each repeating section contains information about each group buffer pool to which this DB2 data sharing member is currently connected.

**Record trace - IFCID 330 - Active Log Space Shortage**

The field labels shown in the following sample layout of “Record Trace - IFCID 330 - Active Log Space Shortage” are described in the following section.

**ACTIVE LOG COPY NUMBER**

Active log copy number (1 or 2).

*Field Name:* QW0330CP

**LAST LOG DATA SET USAGE**

Percentage of the last available active log data set for this log copy that is used.

*Field Name:* QW0330PC
This topic shows detailed information about “Record Trace - IFCID 331 - IBM Service Record”.

This record is for IBM service use.
This topic shows detailed information about “Record Trace - IFCID 332 - IBM Service Record”.

This record is for IBM service use.
This topic shows detailed information about “Record Trace - IFCID 333 - IBM Service Record”.

This record is for IBM service use.
IFCID 335 - System Event Stalled

This topic shows detailed information about “Record Trace - IFCID 335 - System Event Stalled”.

This IFCID records information about stalled system events. These records include the checkpoint process or a log offload task.

Record trace - IFCID 335 - System Event Stalled

The field labels shown in the following sample layout of “Record Trace - IFCID 335 - System Event Stalled” are described in the following section.

STALLED SYST EVT

The stalled system event can be:

CKPT  System checkpoint processor
OFLD  Log offload task

Field Name: QW0335SE

TIMESTAMP

The timestamp of the previous event.

Field Name: QW0335TS

LOG RBA PRIO EVT

The log RBA of the previous event.

Field Name: QW0335PR

CURRENT LOG RBA

The current, highest-written log RBA.

Field Name: QW0335CR
IFCID 342 - WF/TEMP DB Usage

Record Trace - IFCID 342 - WF/TEMP DB Usage

The field labels shown in the following sample layout of “Record Trace - IFCID 342 - WF/TEMP DB Usage” are described in the following section.

WF/TEMP DB USAGE

DATABASE TYPE
The database type.

Field Name: QW0342TY

DBID OF DB
The database ID (DBID).

Field Name: QW0342DB

PSID OF SPACE
The page set object identifier (PSID) of the database.

Field Name: QW0342PS

AGENT TOKEN
The agent token.

Field Name: QW0342AT

CURRENT NO TABLE BLOCKS
The current space for tables that is used by the agent in the database (in KB).

Field Name: QW0342CT

MAX NO TABLE BLOCKS
The maximum space for tables that is used by the agent in the database (in KB).

Field Name: QW0342MT

PARENT TOKEN
The parent token.

Field Name: QW0342PT

CURRENT NO INDEX BLOCKS
The current space for indexes on the tables used by the agent in the database (in KB).
MAX NO INDEX BLOCKS

The maximum space for indexes on the tables used by the agent in the database (in KB).

Field Name: QW0342MI
IFCID 343 - MAXTEMPS Limit/Exceeded

This topic shows detailed information about “Record Trace - IFCID 343 - MAXTEMPS Limit/Exceeded”.

This record is written if the MAXTEMPS zparm limit for an agent is exceeded.

Record trace - IFCID 343 - MAXTEMPS Limit/Exceeded

The field labels shown in the following sample layout of “Record Trace - IFCID 343 - MAXTEMPS Limit/Exceeded” are described in the following section.

AUTH. ID
The authorization ID for the agent.
Field Name: QW0343ID

PACKAGE COLLECTION ID
The package collection ID for the agent.
Field Name: QW0343PC

PACKAGE NAME
The package name for the agent.
Field Name: QW0343PK

PLAN NAME
The plan name for the agent.
Field Name: QW0343PL

MAX TEMP STORAGE
The MAXTEMPS zparm value for the agent (KB).
Field Name: QW0343MS

CURRENT WORKFILE SIZE
The current total system wide usage of WORKFILE storage (KB).
Field Name: QW0343CU

MAX WORKFILE SIZE
The maximum total system wide usage of WORKFILE storage (KB).
Field Name: QW0343MU
IFCID 345 - Trace Data / SP/UDF

This topic shows detailed information about “Record Trace - IFCID 345 - Trace Data / SP/UDF”.

Record trace - IFCID 345 - Trace Data / SP/UDF

The field labels shown in the following sample layout of “Record Trace - IFCID 345 - Trace Data / SP/UDF” are described in the following section.

TRACE BUFFER LENGTH

Length of the trace buffer.

Field Name: QW0345TR_LEN

TRACE BUFFER

Trace buffer.

Field Name: QW0345TR
IFCID 346 - Package/DBRM Detail

This topic shows detailed information about “Record Trace - IFCID 346 - Package/DBRM Detail”.

Record trace - IFCID 346 - Package/DBRM Detail

The field labels shown in the following sample layout of “Record Trace - IFCID 346 - Package/DBRM Detail” are described in the following section.

ACE : 3807507019

ACE

ACE token. You can use this value to correlate this record with other monitor trace records.

Field Name: QW0346_ACE
This topic shows detailed information about “Record Trace - IFCID 350 - SQL Statement”.

IFCID 350 records the complete text of a parsed SQL statement. These records are written when a static or dynamic SQL statement is bound.

**Record trace - IFCID 350 - SQL Statement**

The field labels shown in the following sample layout of “Record Trace - IFCID 350 - SQL Statement” are described in the following section.

```plaintext
OPTIONS : X'04'  HOST LANG : N/A
SQL SEGMENT : PORTION  STMT ID : N/A
STMT TYPE : N/A  CCSID : N/A
SQL LENGTH: 0
SQL STATEMENT:
```

**OPTIONS**

Shows the parser options and host language.

*Field Name:* QW0350OT

**HOST LANG**

Determines the host language. It can have the following values:

- ASSEMBLER
- COBOL
- C
- FORTRAN
- PL/I
- COBOL II
- IBM COBOL
- C++

*Field Name:* QW0350HL

**SQL SEGMENT**

Shows the first, the last, the complete, or a portion of the SQL statement.

*Field Name:* QW0350FG

**STMT ID**

The statement identifier.

*Field Name:* QW0350SI

**STMT TYPE**

The statement type. Possible values are DYNAMIC, STATIC, or N/P.

*Field Name:* QW0350TY

**CCSID**

The coded character set identifier (CCSID).

*Field Name:* QW0350CC

**SQL LENGTH**
IFCID 350 - SQL Statement

The total length of the SQL statement. The maximum length is 5000 bytes.

Field Name: QW0350TL

SQL STATEMENT

Shows the complete SQL statement that is being parsed or only a part of it.

Note: Host variables in this field are represented by :h.

Field Name: QW0350SP
This topic shows detailed information about “Record Trace - IFCID 351 - Wait TCPIP LOB”.

IFCID 0351 is generated at the beginning of the time that is spent waiting for TCP/IP to materialize a LOB. You can activate this trace by starting accounting trace class 3 or 8.

Record trace - IFCID 351 - Wait TCPIP LOB

The field labels shown in the following sample layout of “Record Trace - IFCID 351 - Wait TCPIP LOB” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>REQUEST TYPE</td>
<td>The request type. Possible values are:</td>
</tr>
<tr>
<td>GET</td>
<td>Materialize a LOB/XML value into the database.</td>
</tr>
<tr>
<td>CLOSE</td>
<td>Receive and discard a LOB/XML value from the network.</td>
</tr>
<tr>
<td>CLOSEALL</td>
<td>Receive and discard all the LOB/XML values for this request.</td>
</tr>
</tbody>
</table>

Field Name: QW0351RT
This topic shows detailed information about “Record Trace - IFCID 353 - IBM Service Record”.

This record is for IBM service use.
This topic shows detailed information about “Record Trace - IFCID 354 - IBM Service Record”.

This record is for IBM service use.
IFCID 357 - Beginning of an Index I/O Parallel INSERT

This topic shows detailed information about “Record Trace - IFCID 357 - Beginning of an Index I/O Parallel INSERT”.

IFCID 357 records the beginning of data insertion into an index, with I/O parallelism. This record is written when performance trace class 4 is active.

Record trace - IFCID 357 - Beginning of an Index I/O Parallel INSERT

The field labels shown in the following sample layout of “Record Trace - IFCID 357 - Beginning of an Index I/O Parallel INSERT” are described in the following section.

DATABASE ID: DB123456 TABLE OBID: TB12345678 IDX SPACE PG SET ID: x'1234'

DATABASE ID
The database identifier (ID) of the first index involved in the I/O parallel INSERT.
Field Name: QW0357DB

TABLE OBID
The object identifier (OBID) of the table involved in the INSERT.
Field Name: QW0357TB

IDX SPACE PG SET ID
The page set ID of the first index.
Field Name: QW0357PS
IFCID 358 - End of an Index I/O Parallel INSERT

This topic shows detailed information about “Record Trace - IFCID 358 - End of an Index I/O Parallel INSERT”.

IFCID 358 records the end of data insertion into an index, with I/O parallelism. This record is written when performance trace class 4 is active.

Record trace - IFCID 358 - End of an Index I/O Parallel INSERT

The field labels shown in the following sample layout of “Record Trace - IFCID 358 - End of an Index I/O Parallel INSERT” are described in the following section.

DATABASE ID

The database ID of the last index involved in the I/O parallel INSERT.

Field Name: QW0358DB

TABLE OBID

The object identifier (OBID) of the table involved in the INSERT.

Field Name: QW0358TB

IDX SPACE PG SET ID

The page set ID of the last index.

Field Name: QW0358PS

PARALL. DEGREE

The degree of I/O parallelism on the INSERT.

Field Name: QW0358DE
IFCID 359 - Index Page Split

This topic shows detailed information about “Record Trace - IFCID 359 - Index Page Split”.

IFCID 359 records information about index page splits. This record is written when performance trace class 4 is active.

Record trace - IFCID 359 - Index Page Split

The field labels shown in the following sample layout of “Record Trace - IFCID 359 - Index Page Split” are described in the following section.

DATABASE ID
The database ID of the splitting index.
Field Name: QW0359DB

INDEX PAGE SET ID
The page set ID of the splitting index.
Field Name: QW0359OB

PARTITION NUMBER
The partition number of the splitting index.
Field Name: QW0359PT

SPLITTING PAGE NUMBER
The page number of the splitting index.
Field Name: QW0359PG

FLAGS
Shows if the index depends on the group buffer pool (GBP).
Field Name: QW0359FL

TIMESTAMP BEGIN
The time stamp when index splitting started.
Field Name: QW0359TS

TIMESTAMP END
The time stamp when index splitting stopped.
Field Name: QW0359TE
THIS TOPIC SHOWS DETAILED INFORMATION ABOUT “RECORD TRACE - IFCID 361 - AUDIT ADMIN AUTHORITIES”.

RECORD TRACE - IFCID 361 - AUDIT ADMIN AUTHORITIES

The field labels shown in the following sample layout of “Record Trace - IFCID 361 - Audit Admin Authorities” are described in the following section.

AUTHORITY TYPE

The authority type.

Possible values are:

- SYSDBADM (System DBADM)
- DBCTRL
- DBADM
- SECADM
- ACCSCTRL (ACCESSCTRL)
- SYSADMI (Installation SYSADM)
- SQLADM
- SYSCTRL
- DBMAINT
- SYSOPR
- PACKADM
- SYSOPRI (Installation SYSOPR)
- SYSADM
- DATAACCS (DATAACCESS)
- USER

Field Name: QW0361AT

AUTHID TYPE

The authorization ID type. Possible values are:

‘blank’ Indicates that the authorization ID (AUTH ID) is used.

‘L’ Indicates that ROLE is used.

Field Name: QW0361IT

PRIVILEGE CHECKED

The privilege that was checked. Possible values are provided in the DB2 macro DSNDQW05.

Field Name: QW0361PR

OBJECT TYPE
IFCID 361 - Audit Admin Authorities

The DB2 object type.
Possible values are:
• ACEE
• BUFFER (Bufferpool)
• COLLECT (Collection)
• DATABASE
• DISTINCType (Distinct Type)
• FUNCTION
• SESSIONV (Session Variable)
• JAR
• PACKAGE
• ROLE
• SCHEMA
• Trusted Context
• PROCEDUR (Procedure)
• APPLPLAN (Application Plan)
• LOBTS (LOB Tablespace)
• STOGROUP (Storage Group)
• TAB/VIEW (Table or View)
• USERAUTH (User Auth)
• SEQUENCE
• ROW

Field Name: QW0361OT

AUTHORIZATION ID
The authorization ID or the role that has the authority.
Field Name: QW0361ID

SOURCE OBJ QUALIF
The source object qualifier or owner.
Field Name: QW0361SC

SOURCE OBJ NAME
The source object name.
Field Name: QW0361SN

TARGET OBJ QUALIF
The target object qualifier or owner.
Field Name: QW0361TC

TARGET OBJ NAME
The target object name.
Field Name: QW0361TN

OTHER OBJ NAME
The other object name or subsystem parameter.
Field Name: QW0361ON
IFCID 361 - Audit Admin Authorities

SQL STATEMENT

The SQL statement (truncated at 4000 bytes).

Field Name: QW0361SQ
IFCID 362 - Start Trace and Stop Trace with Audit Policy

Record trace - IFCID 362 - Start Trace and Stop Trace with Audit Policy

The field labels shown in the following sample layout of “Record Trace - IFCID 362 - Start Trace and Stop Trace with Audit Policy” are described in the following section.

STATUS: FAILED TYPE: 15138852 REASON CODE: x'00'

DB2 START UP: N/P DATABASE NAME: N/P

AUDIT CATEGORIES:
  CHECKING X'00'
  VALIDATE X'00'
  OBJMAINT X'00'
  EXECUTE X'00'
  CONTENT X'00'
  SECMAINT X'00'

AUDIT POLICY NAME: N/P
TABLE SCHEMA NAME: N/P
TABLE NAME: N/P
SYSADM CAT VALUES: N/P
DBADM CAT VALUES: N/P
COLLECTION ID: N/P

TABLE NAMES: N/P

STATUS
  Status.
  Possible values are:
  S  Success
  F  Failed
  Field Name: QW0362ST

TYPE
  The type. It can have a value of 'S' for Start Trace.
  Field Name: QW0362TY

REASON CODE
  The reason code.
  Field Name: QW0362RN

DB2 START UP
  The DB2 start up.
  Field Name: QW0362DS

DATABASE NAME
  The database name.
  Field Name: QW0362DB

CHECKING
  The CHECKING category.
  Field Name: QW0362CH

VALIDATE
  The VALIDATE category.
IFCID 362 - Start Trace and Stop Trace with Audit Policy

Field Name: QW0362VA

OBJMAINT

The OBJMAINT category.

Field Name: QW0362OB

EXECUTE

The EXECUTE category.

Field Name: QW0362EX

CONTENT

The CONTEXT category.

Field Name: QW0362CX

SECMAINT

The SECMAINT category.

Field Name: QW0362SM

AUDIT POLICY NAME

The audit policy name.

Field Name: QW0362AP

TABLE SCHEMA NAME

The table schema name.

Field Name: QW0362TS

TABLE NAME

The table name.

Field Name: QW0362TB

SYSADM CAT VALUES

The SYSADMIN category values.

Field Name: QW0362SA

DBADM CAT VALUES

The DBADMIN category values.

Field Name: QW0362DA

COLLECTION ID

The collection ID.

Field Name: QW0362CO

TABLE NAMES

The list of table names traced up to 4K bytes. Each table name is mapped to field QW0362TN_Var.

Field Name: QW0362TN
IFCID 363 consists of the following data sections: QW0363 and QW0363E.

This topic shows detailed information about "Record Trace - IFCID 363 - Data Section QW0363".

This topic shows detailed information about "Record Trace - IFCID 363 - Data Section QW0363E".
IFCID 363 - Data Section QW0363

This topic shows detailed information about “Record Trace - IFCID 363 - Data Section QW0363”.

Record trace - IFCID 363 - Data Section QW0363

The field labels shown in the following sample layout of “Record Trace - IFCID 363 - Data Section QW0363” are described in the following section.

LOCATION NAME

The location name.

Field Name: QW0363LN

PACKAGE NAME

The package name.

Field Name: QW0363PC

PROGRAM NAME

The program name.

Field Name: QW0363PN

CONSISTENCY TOKEN

The time stamp.

Field Name: QW0363TS

STATEMENT NO

The statement number.

Field Name: QW0363SN

QUERY BLOCK NO

The query block number.

Field Name: QW0363QN

PAR. GROUP NO

The parallel group number.

Field Name: QW0363GN

PLANNED DEGREE

The planned (bind time) degree.

Field Name: QW0363BD

PARTITION KIND

The partition kind of the parallel.
IFCID 363 - Data Section QW0363

Field Name: QW0363RK
RECORDS ORDER
The record order: descending or ascending.
Field Name: QW0363OD

IN MEM WORKFILE
Record in memory work file.
Field Name: QW0363IW

INPUT RID IN WKF
Input RID in work file.
Field Name: QW0363RI

OUTPUT RID IN WKF
Output RID in work file.
Field Name: QW0363RO

TOTAL INPT ELEM #
The total number of elements.
Field Name: QW0363NE

TOTAL RECORDS #
The total number of records.
Field Name: QW0363NR

ACTUAL USED WIOEs
Number of actual elements.
Field Name: QW0363AE

PIPE DEGREE
The pipe degree.
Field Name: QW0363PD

PIPE CREATION
The start time of the pipe.
Field Name: QW0363PS

PIPE TERMINATION
The end time of the pipe.
Field Name: QW0363PT

PIPE ELAPSED
The time elapsed between the start and end time of the pipe.
Field Name: RT0363PE
IFCID 363 - Data Section QW0363E

This topic shows detailed information about “Record Trace - IFCID 363 - Data Section QW0363E”.

WORKLOAD ELEMENT INDEX
Identifies the number of the workload element (nth one).
Field Name: QW0363IX

TASK NO
The task number of the subpipe index.
Field Name: QW0363PI

SUB-PIPE CREATION
The subpipe start time.
Field Name: QW0363PB

SUB-PIPE TERMINATION
The end time of the subpipe.
Field Name: QW0363PE

SUB-PIPE ELAPSED
The time elapsed between the start and end time of the subpipe.
Field Name: RT0363SE

LOW BOUND PAGE NO
The page number of low bound of logical partition.
Field Name: QW0363LP

HIGH BOUND PAGE NO
The page number of high bound of logical partition.
Field Name: QW0363HP

LOW KEY BUFFER DATA
The low key buffer.
Field Name: QW0363LB

HIGH KEY BUFFER DATA
The high key buffer.
Field Name: QW0363HB

NBR OF ROWS CONSUMED
The number of rows consumed.
Field Name: QW0363NI

NBR OF ROWS PRODUCED
The number of rows produced.
Field Name: QW0363NO

QW0363CN
This field is for IBM service.
IFCID 363 - Data Section QW0363E

Field Name: QW0363CN

QW0363BI
This field is for IBM service.
Field Name: QW0363BI

QW0363EI
This field is for IBM service.
Field Name: QW0363EI
IFCID 365 - Remote Location Statistics

This topic shows detailed information about “Record Trace - IFCID 365 - Remote Location Statistics”.

IFCID 0365 records detailed statistics about the remote locations with which a DB2 subsystem communicates using the DRDA protocol. This record is written when Statistics trace class 7 is on. The DDF DATA BY LOCATION section is shown for each location with which the DB2 subsystem communicates.

Record trace - IFCID 365 - Remote Location Statistics

The field labels shown in the following sample layout of “Record Trace - IFCID 365 - Remote Location Statistics” are described in the following section.

REMOTE LOCATIONS STATISTICS

SECTIONS IN RECORD ........: 3 ANOTHER IFCID365 FOLLOWS ..: NO

DDF DATA BY LOCATION

LOCATION NAME (SHORT).......: ::FFFF:192.0.1.6 PRDID REMOTE LOCATION .....: JCC03610
LOCATION NAME (LONG).......: ::FFFF:192.0.1.68

INITIATED CONVERSATIONS....: 0 DEALLOCATED CONVERSATIONS..: 0
MESSAGES SENT TO REMOTE....: 10815873 MESSAGES RECV FR REMOTE.....: 10815887
SQL STMTS SENT TO REMOTE...: 0 SQL STMTS RECV FR REMOTE...: 10540296
BYTES SENT TO REMOTE.......: 2005208369 BYTES RECV FR REMOTE......: 1711891973
ROWS SENT TO REMOTE........: 3612785 ROWS RETRIEVED FR REMOTE...: 0
BLOCKS TRANSMITTED.........: 3684736 BLOCKS RECEIVED............: 0

SECTIONS IN RECORD

Provides the number of QLST sections that are part of this IFCID 365 record.
Field Name: QW0365NO

ANOTHER IFCID365 FOLLOWS

This bitcounter indicates if another IFCID 365 is shown in the DB2 trace data.
Field Name: QW0365FL

LOCATION NAME (SHORT)

The name of the remote location.
Field Name: QLSTLOCN

PRDID REMOTE LOCATION

The product ID and version of the remote location.
Field Name: QLSTPRID

LOCATION NAME (LONG)

The name of the remote location.
Field Name: QLSTLOCN

INITIATED CONVERSATIONS

The number of conversations that were initiated from the requester location. This value is maintained at the requester location.
A conversation is a specific instance of using TCP/IP or SNA LU 6.2 to transfer information between a requester and a server. A conversation is a logical connection between a requester and a server.

Field Name: QLSTCNVS

DEALLOCATED CONVERSATIONS

The number of conversations that were deallocated from this site to the remote site.

Field Name: QLSTCNVT

INITIATED FROM REMOTE SITE

The number of conversations that were initiated from the requester to the server location. This value is updated at the server location.

Field Name: QLSTCNVR

MESSAGES SENT TO REMOTE

The number of messages sent to the remote location. A message is a group of characters and control bit sequences transferred on a single TCP/IP or SNA API call. This value is maintained at the location where the messages originated.

Field Name: QLSTMSGS

MESSAGES RECV FR REMOTE

The number of messages received by VTAM from the remote location. This value is maintained at the location where the messages were received.

More messages might be sent from the server location than are received by the requester due to the manner in which distributed SQL statements are processed internally.

Field Name: QLSTMSGR

SQL STMTS SENT TO REMOTE

The number of SQL statements sent to the remote server. This value is updated at the requester location.

Field Name: QLSTSQLS

SQL STMTS RECV FR REMOTE

The number of SQL statements received from the requester location. This value is updated at the server location.

Field Name: QLSTSQLR

BYTES SENT TO REMOTE

The number of bytes of data sent to the requester location. This value is maintained at the server location.

Field Name: QLSTBYTES

BYTES RECV FR REMOTE

The number of bytes of data received from the server location. This value is maintained at the requester location.

More bytes of data might be sent from the server location than are received by the requester due to the manner in which distributed SQL statements are processed internally.
Field Name: QLSTBYTR
ROWS SENT TO REMOTE
   The number of data rows sent to the requester location (includes SQLDA).
   This value is updated at the server location.
Field Name: QLSTROWS
ROWS RETRIEVED FR REMOTE
   The number of data rows received from the server location. This value is
   maintained at the requester location.

Note:
   • This value does not include any SQLDA or SQLCA transmitted.
   • Block fetch can significantly affect the number of rows sent across the
     network. When used with nonupdate cursors, block fetch groups as
     many rows as possible into the message buffer, and transmits the buffer
     over the network without requiring a VTAM message. Consequently,
     more rows of data might be sent from the server location than are
     received by the requester location.
     This is especially true when DB2 private protocol is used because
     multiple blocks can be transmitted from the server with no intervening
     messages from the requester.
Field Name: QLSTROWR
BLOCKS TRANSMITTED
   The number of blocks transmitted using block fetch. This value is
   maintained at the server location.
Field Name: QLSTBTBF
BLOCKS RECEIVED
   The number of blocks received from the remote location using block fetch.
   This value is maintained at the requester location.
Field Name: QLSTBRBF
IFCID 366 - Incompatible Function Char(DEC) Executed

This topic shows detailed information about “Record Trace - IFCID 366 - Incompatible Function Char(DEC) Executed”.

Record trace - IFCID 366 - Incompatible Function Char(DEC) Executed

The field labels shown in the following sample layout of “Record Trace - IFCID 366 - Incompatible Function Char(DEC) Executed” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>COLLECTION ID</td>
<td>The package collection ID.</td>
</tr>
<tr>
<td>PROGRAM NAME</td>
<td>The program name.</td>
</tr>
<tr>
<td>CONTOKEN (TS)</td>
<td>The consistency token is shown in hexadecimal format.</td>
</tr>
<tr>
<td>INDIC CHAR(DEC)</td>
<td>The function type is incompatible. It can have the following values:</td>
</tr>
<tr>
<td></td>
<td>1 Indicates that the pre-version 10 CHAR built-in function has been invoked.</td>
</tr>
<tr>
<td></td>
<td>2 Indicates that the pre-version 10 VARCHAR built-in function has been invoked.</td>
</tr>
<tr>
<td></td>
<td>3 Indicates that an unsupported character representation of a timestamp string was used.</td>
</tr>
<tr>
<td></td>
<td>8 Indicates that the DB2 for z/OS server returned output data which matches the data types of the corresponding CALL statement arguments when DDF_COMPATIBILITY zparm is set to SP_PARMS_NJV.</td>
</tr>
<tr>
<td>STMT NBR QUERY</td>
<td>The statement number of the query.</td>
</tr>
<tr>
<td>PLAN NAME QUERY</td>
<td>The plan name of the query.</td>
</tr>
</tbody>
</table>
IFCID 366 - Incompatible Function Char(DEC) Executed

STMT ID
The statement identifier.
Field Name: QW0366SI

STMT TYPE
The statement information. It can be a DYNAMIC or STATIC statement.
Field Name: QW0366TY

SECTION
The section number.
Field Name: QW0366SE

VERSION LENGTH
The version length.
Field Name: QW0366VL

VERSION
The version.
Field Name: QW0366VN
IFCID 369 - Aggregated Accounting Statistics

This topic shows detailed information about “Record Trace - IFCID 369 - Aggregated Accounting Statistics”.

IFCID 369 contains aggregated Accounting data listed by connection type. It is shown at Statistics intervals, for more information refer to “Aggregated Accounting Statistics” on page 49-10.

The following data sections provide information for each connection type that is listed in DATA SECTION 2 - CONNECTION TYPES:

- DATA SECTION 3: ACCOUNTING DATA - “IFCID 003 - Instrumentation Accounting Data” on page 40-175
- DATA SECTION 4: ACCOUNTING DATA OVERFLOW - “IFCID 003 - Instrumentation Accounting Data Overflow” on page 40-190

Record trace - IFCID 369 - Aggregated Accounting Statistics

The field labels shown in the following sample layout of “Record Trace - IFCID 369 - Aggregated Accounting Statistics” are described in the following section.

LOCATION: DDFD6YO OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-1
GROUP: N/P RECORD TRACE - LONG REQUESTED FROM: NOT SPECIFIED
MEMBER: N/P TO: NOT SPECIFIED
SUBSYSTEM: D6Y0 ACTUAL FROM: 07/10/12 07:18:00.09
DB2 VERSION: V10 PAGE DATE: 07/10/12
PRIMAUTH CONNECT INSTANCE END_USER WS_NAME TRANSACT ORIGAUTH CORRNAME CONNTYPE RECORD TIME DESTNO ACE IFC DESCRIPTION DATA
PLANNAME CORRNMBR TCB CPU TIME ID
-------- -------- ----------- ----------------- ------ --- --- -------------- ------------------------------------------------------
SYSOPR D6Y0 C9072C42371F 'BLANK' 'BLANK' 'BLANK'
SYSOPR 016.WVSM 'BLANK' 07:18:00.09381935 41903 1 369 AGGR ACCOUNTNG
NETWORKID: D6Y0 LUNAME: D6Y0 LUWSEQ: 1
'TBLANK' T 01 N/P STATISTICS
TIMESTAMP WHEN ENABLED: 07/10/13 07:14:32.981817
TIMESTAMP WHEN DISABLED: 07/09/13 14:50:57.335291

DATA SECTION 2: CONNECTION TYPES

- CONNECTION TYPE 1: UTILITY
- CONNECTION TYPE 2: BATCH
- CONNECTION TYPE 3: RRSAF ATTACH
- CONNECTION TYPE 4: DDF CONNECTION

DATA SECTION 3: ACCOUNTING DATA

ACCOUNTING DATA FOR TYPE: 1
INSTRUMENTATION ACCOUNTING DATA
CLASS 1 BEGINNING STORE CLOCK TIME 01/02/00 00:00:00 19.770757
ELAPSED TIME 19.770757 MVS TCB TIME 1.383040
BEGINNING MVS TCB TIME 0.000000 ENDING MVS TCB TIME 1.383040
STORED PROC ELAPSED TIME 0.000000 CONVERSION FACTOR 0
STORED PROCEDURE TCB TIME 0.000000 PAR.TASKS: 97 PARTOKEN: 'X'00000000'
UOP ELAPSED TIME 0.000000 COMMTX: 5090 SYPT REL.: 0
UOP TCB TIME 0.000000 ROLLBACKS: 1 SYPT RLB.: 0
UOP NF SE CPU TIME 0.000000
UOP NF ELAPSED TIME 0.000000
UOP NF CP CPU TIME 0.000000
CLASS 1/2 STORED PROC ZIP TCB TIME 0.000000
STORED PROC ELAPSED TIME 0.000000
STORED PROC CP ELAPSED TIME 0.000000
UOP NF SE CPU TIME 0.000000
UOP NF ELAPSED TIME 0.000000
UOP NF CP CPU TIME 0.000000
CLASS 2 DB2 ELAPSED TIME 5.757953 DB2 ENTRY/EXIT EVENTS 2512
TCB TIME 0.448220 NON-ZERO CLASS 2 YES
STORED PROC ELAPSED TIME 0.000000 CLASS 2 DATA COLLECTED YES
STORED PROCEDURE TCB TIME 0.000000 STORED PROC, ENTRY/EXITS 0
UOP ELAPSED TIME 0.000000 UOF SQL ENTRY/EXIT EVENTS 0
CP CPU TIME 0.000000 SE CPU TIME 1.548725
TRIG ELAP TIME UNDER ENCLAVE 0.000000 SE ELIGIBLE CP CPU TIME 903 22:07:20.834
TRIG TCB TIME UNDER ENCLAVE 0.000000 QMCTRTI ZIIP
TRIG ELAP TIME NOT UNDER ENCLAVE 0.000000
TRIG TCB TIME NOT UNDER ENCLAVE 0.000000
IFCID 369 - Aggregated Accounting Statistics

CLASS 3 LOCK/LATCH(DB2+IRLM) SUSP TIME N/A LOCK/LATCH(DB2+IRLM) SUSP EVENTS N/A
WAIT TIME LOCAL LOCKS 0.013750 LOCAL LOCK WAIT TRACE EVENTS 18

LOCATION: DDFD6Y0
GROUP: N/P
MEMBER: N/P
SUBSYSTEM: D6Y0
DB2 VERSION: V10

WAIT TIME LOCAL LOCKS 0.013750 LOCAL LOCK WAIT TRACE EVENTS 18

GROUP: N/P RECORD TRACE - LONG
REQUESTED FROM: NOT SPECIFIED
TO: NOT SPECIFIED

MEMBER: N/P

SUBSYSTEM: D6Y0
ACTUAL FROM: 07/10/13 07:18:00.09
DB2 VERSION: V10
PAGE DATE: 07/10/13

PRIMAUTH CONNECT INSTANCE END_USER WS_NAME TRANSACT
ORIGAUTH CORRNAME CONNTYPE RECORD TIME DESTNO ACE IFC DESCRIPTION DATA
PLANNAME CORRNMBR TCB CPU TIME ID
-------- -------- ----------- ----------------- ------ --- --- -------------- ------------------------------------------------------
SYSOPR D6Y0 'BLANK' 'BLANK' 'BLANK' 'BLANK' T 01 N/P STATISTICS
07:18:00.09381935 41903 1 369 AGGR ACCOUNTING
'BLANK' 016.WVSM 'BLANK' 'BLANK' 'BLANK' 'BLANK' T 01 N/P STATISTICS
07:18:00.09381935 41903 1 369 AGGR ACCOUNTING

SYSDPR
016.WVSM 'BLANK' 'BLANK' 'BLANK' 'BLANK' T 01 N/P STATISTICS

SYSDPR
016.WVSM 'BLANK' 'BLANK' 'BLANK' 'BLANK' T 01 N/P STATISTICS

ACCOUNTING DATA FOR TYPE: 2

ACCOUNTING DATA FOR OVERFLOW TYPE: 1

ACCOUNTING DATA FOR OVERFLOW TYPE: 2

DATA SECTION 4: ACCOUNTING DATA OVERFLOW

TIMESTAMP WHEN ENABLED
The timestamp shows when the IFCID 369 statistics collection was enabled.

Field Name: QW0369ST

TIMESTAMP WHEN DISABLED
The timestamp shows when the IFCID 369 statistics collection was disabled.

Field Name: QW0369SP

Chapter 40. IFCID Record Blocks 40-831
CONNECTION TYPE I (for $I = 1,\ldots,6$)

The connection name for which Accounting data has been aggregated.

Field Name: QW0369CN
IFCID 370 - Database Open Information

This topic shows detailed information about “Record Trace - IFCID 370 - Database Open Information”.

Record trace - IFCID 370 - Database Open Information

The field labels shown in the following sample layout of “Record Trace - IFCID 370 - Database Open Information” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATA SET NAME</td>
<td>QW0370DN</td>
</tr>
<tr>
<td>FLAGS</td>
<td>X'00'</td>
</tr>
<tr>
<td>ACE ADDRESS</td>
<td>X'1602D430'</td>
</tr>
<tr>
<td>DATABASE ID</td>
<td>6</td>
</tr>
<tr>
<td>OBID</td>
<td>101</td>
</tr>
<tr>
<td>PART NUMBER</td>
<td>X'00000001'</td>
</tr>
<tr>
<td>INSTANCE NUMBER</td>
<td>X'00000001'</td>
</tr>
<tr>
<td>DSMAX</td>
<td>500</td>
</tr>
<tr>
<td>OPENED DATA SETS</td>
<td>13</td>
</tr>
<tr>
<td>ALLOCATION TIME</td>
<td>0.001435</td>
</tr>
<tr>
<td>OPEN TIME</td>
<td>0.020216</td>
</tr>
</tbody>
</table>

DATA SET NAME

The data set name.

Field Name: QW0370DN

FLAGS

The flags.

Field Name: QW0370FG

ACE ADDRESS

The address of the agent control element (ACE).

Field Name: QW0370AC

DATABASE ID

The database ID (DBID).

Field Name: QW0370DB

OBID

The page set OBID.

Field Name: QW0370OB

PART NUMBER

The part number.

Field Name: QW0370PN

INSTANCE NUMBER

The instance number.

Field Name: QW0370IN

DSMAX

The maximum number of data sets (DSMAX).

Field Name: QW0370DM

OPENED DATA SETS

The number of opened data sets.

Field Name: QW0370DO

ALLOCATION TIME
IFCID 370 - Database Open Information

The allocation time. It is based on the execution time of SVC 99 invoked by DSNB1OST.

Field Name: QW0370AL

OPEN TIME

The open time. It is based on the execution time of DSNB4ODS.

Field Name: QW0370OP
IFCID 371 - Database Close Information

This topic shows detailed information about “Record Trace - IFCID 371 - Database Close Information”.

Record trace - IFCID 371 - Database Close Information

The field labels shown in the following sample layout of “Record Trace - IFCID 371 - Database Close Information” are described in the following section.

DATA SET NAME
The data set name.
Field Name: QW0371DN

FLAGS
The flags.
Field Name: QW0371FG

ACE ADDRESS
The address of the agent control element (ACE).
Field Name: QW0371AC

DATABASE ID
The database ID (DBID).
Field Name: QW0371DB

OBID
The page set OBID.
Field Name: QW0371OB

PART NUMBER
The part number.
Field Name: QW0371PN

INSTANCE NUMBER
The instance number.
Field Name: QW0371IN

DSMAX
The maximum number of data sets (DSMAX).
Field Name: QW0371DM

DEALLOCATION TIME
The deallocation time. It is based on the execution time of SVC 99 invoked by DSNB1CST.
Field Name: QW0371DA

CLOSE TIME

Chapter 40. IFCID Record Blocks 40-835
IFCID 371 - Database Close Information

The close time. It is based on the execution time of DSNB4CDS.

**Field Name:** QW0371CL

**OPENED DATA SETS**

The number of opened data sets.

**Field Name:** QW0371DO
IFCID 377 - Pseudo Delete Daemon Cleanup

This topic shows detailed information about “Record Trace - IFCID 377 - Pseudo Delete Daemon Cleanup”.

IFCID 0377 records automatic cleanup of pseudo-deleted index entries by the index pseudo-delete daemon. This record is not associated with a trace class.

Record trace - IFCID 377 - Pseudo Delete Daemon Cleanup

The field labels shown in the following sample layout of “Record Trace - IFCID 377 - Pseudo Delete Daemon Cleanup” are described in the following section.

DATABASE ID

The database ID of the index for which entries are cleaned up.

Field Name: QW0377DB

INDEX PAGE NUMBER

The page number of the index page that was cleaned up.

Field Name: QW0377PG

PARTITION NUMBER

The index partition number.

Field Name: QW0377PT

INDEX PAGE SET ID

The page set ID of the index.

Field Name: QW0377OB

PD ENTRIES REMOVED

The number of pseudo-deleted entries that were removed.

Field Name: QW0377NU

FLAG

The reason for the page removal:

- A page is deleted from an index (DB2 field: QW0377DL)
- A page is cleaned up (DB2 field: QW0377CL)

Field Name: RW0377DL
IFCID 378 - Accel. Call Event Begin (DB2 10 or later)

This topic shows detailed information about “Record Trace - IFCID 378 - Accel. Call Event Begin (DB2 10 or later)”.

Record trace - IFCID 378 - Accel. Call Event Begin (DB2 10 or later)

The field labels shown in the following sample layout of “Record Trace - IFCID 378 - Accel. Call Event Begin (DB2 10 or later)” are described in the following section.

ACCELERATOR NAME

The name of the accelerator.

**Field Name:** QW0378ACN
IFCID 379 - Accel. Call Event End

This topic shows detailed information about “Record Trace - IFCID 379 - Accel. Call Event End (DB2 10 or later)”.

Record trace - IFCID 379 - Accel. Call Event End (DB2 10 or later)

The field labels shown in the following sample layout of “Record Trace - IFCID 379 - Accel. Call Event End (DB2 10 or later)” are described in the following section.

ACCELERATOR NAME

The name of the accelerator.

Field Name: QW0379ACN
IFCID 380 - Stored Procedure Detail Record

This topic shows detailed information about “Record Trace - IFCID 380 - Stored Procedure Detail Record”.

IFCID 380 (Stored procedure detail record) and IFCID 381 (UDF detail record) have the same mapping structure.

Both records are written at the beginning and the end of a stored procedure or UDF.

The first data section of IFCID 380 shows fields provided with IFCID 233. The second data section starting with CURRENT TOTAL NESTED CLASS 1 CP TIME shows additional fields for IFCID 380 and IFCID 381.

Record trace - IFCID 380 - Stored Procedure Detail Record

The field labels shown in the following sample layout of “Record Trace - IFCID 380 - Stored Procedure Detail Record” are described in the following section.

LOCATION NAME
   The location name.
   **Field Name:** QW0233LN

COLLECTION ID
   The package collection identifier.
   **Field Name:** QW0233PC

PROGRAM NAME
   The program name.
   **Field Name:** QW0233PN

SCHEMA NAME
   The name of the schema associated with this routine.
   **Field Name:** QW0233SC

ROUTINE NAME
   The specific name of the routine.
   **Field Name:** QW0233PR
VERSION NAME
The name of the version.
Field Name: QW0233VER

ROUTINE TYPE
The routine type can have the following values:

PROCEDURE
The routine is a stored procedure

FUNCTION
The routine is a User-Defined Function
Field Name: QW0233TY

CONSISTENCY TOKEN
The consistency token.
Field Name: QW0233TS

ENTRY/EXIT TYPE
The entry or exit event type can have the following values:

ENTERING
The agent is entering a routine.

RETURNED
The agent has returned from a routine.
Field Name: QW0233EX

NESTING LEVEL
The nesting level of the routine.
Field Name: QW0233NL

STATEMENT NO
The statement number of the statement executed.
Field Name: QW0233SN

TYPE
The statement type. Possible values are DYNAMIC or STATIC.
Field Name: QW0233STY

ROUTINE ID
The routine identifier.
Field Name: QW0233RID

STATEMENT ID
The unique identifier of the currently executing statement. It is shown as an integer and in hexadecimal format.
Field Name: QW0233SID

CONV INTO HEX
The unique identifier of the currently executing statement. It is shown as an integer and in hexadecimal format.
FIELD: QW0233SID

CURRENT TOTAL NESTED CLASS 1 CP TIME

The current, total, nested class 1 CP time. This does not include time spent executing on an IBM specialty engine.

FIELD: QW0380_CLS1CP

CURRENT TOTAL NESTED CLASS 1 SE TIME

The current, total, nested class 1 specialty engine time.

FIELD: QW0380_CLS1SE

CURRENT TOTAL NESTED CLASS 2 CP TIME

The current, total, nested class 2 CP time. This is time in the DB2 processing SQL statements. This time also includes in DB2 time needed to connect and disconnect the SP task for non-SQLP stored procedures. This does not include time spent executing on an IBM specialty engine.

FIELD: QW0380_CLS2CP

CURRENT TOTAL NESTED CLASS 2 SE TIME

The current, total, nested class 2 specialty engine time. This is the time in DB2 processing SQL statements.

FIELD: QW0380_CLS2SE

CURRENT TOTAL NESTED ELAPSED CLASS 2 TIME

The current, total, nested elapsed class 2 time. This is the time in DB2 processing SQL statements. This time also includes in DB2 time needed to connect and disconnect the SP task for non-SQLP stored procedures.

FIELD: QW0380_CLS2ELAP
IFCID 381 - UDF Detail Record

This topic shows detailed information about “Record Trace - IFCID 381 - UDF Detail Record”.

IFCID 380 (Stored procedure detail record) and IFCID 381 (UDF detail record) have the same mapping structure.

Both records are written at the beginning and the end of a stored procedure or UDF.

The first data section of IFCID 380 shows fields provided with IFCID 233. The second data section starting with CURRENT TOTAL NESTED CLASS 1 CP TIME shows additional fields for IFCID 380 and IFCID 381.

Record trace - IFCID 381 - UDF Detail Record

The field labels shown in the following sample layout of “Record Trace - IFCID 381 - UDF Detail Record” are described in the following section.

LOCATION NAME

The location name.

Field Name: QW0233LN

COLLECTION ID

The package collection identifier.

Field Name: QW0233PC

PROGRAM NAME

The program name.

Field Name: QW0233PN

SCHEMA NAME

The name of the schema associated with this routine.

Field Name: QW0233SC

ROUTINE NAME

The specific name of the routine.

Field Name: QW0233PR

VERSION NAME
IFCID 381 - UDF Detail Record

The name of the version.
Field Name: QW0233VER

ROUTINE TYPE
The routine type can have the following values:

PROCEDURE
The routine is a stored procedure

FUNCTION
The routine is a User-Defined Function
Field Name: QW0233TY

CONSISTENCY TOKEN
The consistency token.
Field Name: QW0233TS

ENTRY/EXIT TYPE
The entry or exit event type can have the following values:

ENTERING
The agent is entering a routine.

RETURNED
The agent has returned from a routine.
Field Name: QW0233EX

NESTING LEVEL
The nesting level of the routine.
Field Name: QW0233NL

STATEMENT NO
The statement number of the statement executed.
Field Name: QW0233SN

TYPE
The statement type. Possible values are DYNAMIC or STATIC.
Field Name: QW0233STY

ROUTINE ID
The routine identifier.
Field Name: QW0233RID

STATEMENT ID
The unique identifier of the currently executing statement. It is shown as an integer and in hexadecimal format.
Field Name: QW0233SID

CONV INTO HEX
The unique identifier of the currently executing statement. It is shown as an integer and in hexadecimal format.
Field Name: QW0233SID
CURRENT TOTAL NESTED CLASS 1 CP TIME
The current, total, nested class 1 CP time. This does not include time spent executing on an IBM specialty engine.

Field Name: QW0380_CLS1CP

CURRENT TOTAL NESTED CLASS 1 SE TIME
The current, total, nested class 1 specialty engine time.

Field Name: QW0380_CLS1SE

CURRENT TOTAL NESTED CLASS 2 CP TIME
The current, total, nested class 2 CP time. This is time in the DB2 processing SQL statements. This time also includes in DB2 time needed to connect and disconnect the SP task for non-SQLP stored procedures. This does not include time spent executing on an IBM specialty engine.

Field Name: QW0380_CLS2CP

CURRENT TOTAL NESTED CLASS 2 SE TIME
The current, total, nested class 2 specialty engine time. This is the time in DB2 processing SQL statements.

Field Name: QW0380_CLS2SE

CURRENT TOTAL NESTED ELAPSED CLASS 2 TIME
The current, total, nested elapsed class 2 time. This is the time in DB2 processing SQL statements. This time also includes in DB2 time needed to connect and disconnect the SP task for non-SQLP stored procedures.

Field Name: QW0380_CLS2ELAP
IFCID 384 - IBM Service Record

This topic shows detailed information about “Record Trace - IFCID 384 - IBM Service Record”.

This record is for IBM service use.
IFCID 385 - IBM Service Record

This topic shows detailed information about “Record Trace - IFCID 385 - IBM Service Record”.

This record is for IBM service use.
IFCID 386 - IBM Service Record

This topic shows detailed information about “Record Trace - IFCID 386 - IBM Service Record”.

This record is for IBM service use.
This topic shows detailed information about “Record Trace - IFCID 397 - IBM Service Record”.

This record is for IBM service use.
This topic shows detailed information about “Record Trace - IFCID 398 - IBM Service Record”.

This record is for IBM service use.
This topic shows detailed information about “Record Trace - IFCID 399 - IBM Service Record”.

This record is for IBM service use.
### IFCID 401 - Static Statements in EDM Pool

This topic shows detailed information about “Record Trace - IFCID 401 - Static Statements in EDM Pool”.

IFCID 401 has READS and READA capability. It supports threshold value filtering.

**Record trace - IFCID 401 - Static Statements in EDM Pool**

The field labels shown in the following sample layout of “Record Trace - IFCID 401 - Static Statements in EDM Pool” are described in the following section.

<table>
<thead>
<tr>
<th>PACKAGE NAME</th>
<th>CTSH2122</th>
</tr>
</thead>
<tbody>
<tr>
<td>COLLECTION ID</td>
<td>OPMTEST</td>
</tr>
<tr>
<td>DATE/TIME WHEN INSERTED</td>
<td>X'201010418104541179034' STORE CLOCK FORMAT: X'C7A3DB35A929A195'</td>
</tr>
<tr>
<td>DATE/TIME WHEN UPDATED</td>
<td>X'20110418112535104180' STORE CLOCK FORMAT: X'C7A3E420AF8B4109'</td>
</tr>
<tr>
<td>CONSISTENCY TOKEN</td>
<td>X'7844424B71324169'</td>
</tr>
<tr>
<td>STATEMENT IDENTIFIER</td>
<td>132285</td>
</tr>
<tr>
<td>NBR OF EXECUTIONS</td>
<td>50</td>
</tr>
<tr>
<td>NBR OF GETPAGES</td>
<td>0</td>
</tr>
<tr>
<td>NBR OF ROWS PROCESSED</td>
<td>0</td>
</tr>
<tr>
<td>NBR OF INDEX SCANS</td>
<td>0</td>
</tr>
<tr>
<td>NBR OF BUFFER WRITES</td>
<td>0</td>
</tr>
<tr>
<td>ACCUMULATED TIME VALUES SECTION</td>
<td></td>
</tr>
<tr>
<td>IN-DB2 ELAPSED</td>
<td>0.001362</td>
</tr>
<tr>
<td>IN-DB2 CPU</td>
<td>0.001095</td>
</tr>
<tr>
<td>WAIT FOR SYNC I/O</td>
<td>N/P</td>
</tr>
<tr>
<td>WAIT FOR LOCK/LATCH</td>
<td>N/P</td>
</tr>
<tr>
<td>WAIT FOR GLOBAL LOCKS</td>
<td>N/P</td>
</tr>
<tr>
<td>WAIT FOR READ BY OTHER THR</td>
<td>N/P</td>
</tr>
<tr>
<td>WAIT FOR PAGE LATCH</td>
<td>N/P</td>
</tr>
<tr>
<td>WAIT FOR CLAIM RELEASE</td>
<td>N/P</td>
</tr>
<tr>
<td>RID LIST SECTION</td>
<td></td>
</tr>
<tr>
<td>(HJA=HYBRID JOIN APPEND, IA=INDEX ACCESS, OV=OVERFLOW, RL=RID LIST)</td>
<td></td>
</tr>
<tr>
<td>RL NOT USED LIMIT EXCEEDED</td>
<td>0</td>
</tr>
<tr>
<td>RL NOT USED NO STOR AVAIL</td>
<td>0</td>
</tr>
<tr>
<td>RL OV - NO POOL STOR AVAIL</td>
<td>0</td>
</tr>
<tr>
<td>RL OV - RIDS EXCEED LIMIT</td>
<td>0</td>
</tr>
<tr>
<td>HJA - NO POOL STOR AVAIL</td>
<td>0</td>
</tr>
<tr>
<td>HJA - RIDS EXCEED LIMIT</td>
<td>0</td>
</tr>
<tr>
<td>RL RETRIEVAL IA SKIPPED</td>
<td>0</td>
</tr>
</tbody>
</table>

#### PACKAGE NAME

The package name.

**Field Name**: QW0401PK

#### COLLECTION ID

The collection ID.

**Field Name**: QW0401CL

#### DATE/TIME WHEN INSERTED

The date or time when the statement was inserted into the EDM Pool.

**Field Name**: QW0401TM

#### STORE CLOCK FORMAT

The date or time when the statement was inserted into the EDM pool (in store clock format) (DB2 field: QW0401TM2).

**Field Name**: RT401TM2

#### DATA/TIME WHEN UPDATED

The date or time when statement statistics were updated (in external format) (DB2 field: QW0401UT2).

**Field Name**: RT401UT2

#### STORE CLOCK FORMAT
IFCID 401 - Static Statements in EDM Pool

The date or time when statement statistics were updated (in store clock format) (DB2 field: QW0401UT1).

Field Name: RT401UT1

CONSISTENCY TOKEN
The consistency token of the package.

Field Name: QW0401CT

STATEMENT IDENTIFIER
The statement identifier.

Field Name: QW0401ID

NBR OF EXECUTIONS
The number of executions.

Field Name: QW0401EX

NBR OF SYNC BUFFER READS
The number of synchronous buffer reads.

Field Name: QW0401SR

NBR OF GETPAGES
The number of Getpages.

Field Name: QW0401GP

NBR OF ROWS EXAMINED
The number of rows examined.

Field Name: QW0401ER

NBR OF ROWS PROCESSED
The number of rows processed.

Field Name: QW0401PR

NBR OF SORTS
The number of sorts.

Field Name: QW0401ST

NBR OF INDEX SCANS
The number of index scans.

Field Name: QW0401IX

NBR OF TABLESPACE SCANS
The number of tablespace scans.

Field Name: QW0401TB

NBR OF BUFFER WRITES
The number of buffer writes.

Field Name: QW0401WT

NBR OF PAR. GRPS CREATED
The number of parallel groups created.
IFCID 401 - Static Statements in EDM Pool

Field Name: QW0401PG
IN-DB2 ELAPSED
Accumulated in-DB2 elapsed time.
Field Name: QW0401ET
IN-DB2 CPU
The accumulated in-DB2 CPU time. This time includes CPU consumed on an IBM specialty engine.
Field Name: QW0401CP
WAIT FOR SYNC I/O
The accumulated wait time for synchronous I/O.
Field Name: QW0401SI
WAIT FOR LOCK/LATCH
The accumulated wait time for locks.
Field Name: QW0401LK
SYNC EXEC UNIT SWITCH
The accumulated wait time for synchronous execution unit switch.
Field Name: QW0401EU
WT FOR GLOBAL LOCKS
The accumulated wait time for global locks.
Field Name: QW0401GL
WT FOR READ BY OTHER THR
The accumulated wait time for a read activity done by another thread.
Field Name: QW0401OR
WT FOR WRITE BY OTHER THR
The accumulated wait time for a write activity done by another thread.
Field Name: QW0401OW
WAIT FOR LATCH REQ
The accumulated wait time for a latch request.
Field Name: QW0401LH
WAIT FOR PAGE LATCH
The accumulated wait time for a page latch.
Field Name: QW0401PL
WAIT FOR DRAIN LOCK
The accumulated wait time for a drain lock.
Field Name: QW0401DL
WAIT FOR CLAIM RELEASE
The accumulated wait time for a drain lock that is waiting for claims to be released.
Field Name: QW0401CM

WAIT FOR LOG WRITER

The accumulated wait time for a log writer.

Field Name: QW0401LW

RL NOT USED LIMIT EXCEEDED

The number of times RID list was not used because the number of:
- RIDs would have exceeded the DB2 limits
- RID blocks exceeded the value set by the MAXTEMPS_RID system parameter

Field Name: QW0401RL

RL NOT USED NO STOR AVAIL

The number of time a RID list was not used because there is not enough storage available to hold the list of RIDs. This also applies if the work file storage was not available.

Field Name: QW0401RS

RL OV - NO POOL STOR AVAIL

The number of times a RID list was overflowed to a work file because no RID pool storage was available to hold the list of RIDs (DB2 field: QW0401WFRIDS).

Field Name: RT401IDS

RL OV - RIDS EXCEED LIMIT

The number of times a RID list was overflowed to a work file because the number of RIDs exceeded one or more internal limits (DB2 field: QW0401WFRIDT).

Field Name: RT401IDT

HJA - NO POOL STOR AVAIL

The number of times a RID list append for a hybrid join was interrupted because no RID pool storage was available to hold the list of RIDs. It shows the number of times DB2 interrupted the RID phase and switched to the data phase (DB2 field: QW0401HJINCS).

Field Name: RT401NCS

HJA - RIDS EXCEED LIMIT

The number of times a RID list append for a hybrid join was interrupted because the number of RIDs exceeded one or more internal limits. It shows the number of times DB2 interrupted the RID phase and switched to the data phase (DB2 field: QW0401HJINCT).

Field Name: RT401NCT

RL RETRIEVAL IA SKIPPED

The number of times a RID list retrieval for multiple index access was skipped because DB2 predetermined the outcome of index ANDing or ORing (DB2 field: QW0401RSMIAP).

Field Name: RT401IAP
IFCID 402 - System Profile - Monitoring Statistics

IFCID 402 - System Profile - Monitoring Statistics (DB2 10 or later)

This topic shows detailed information about “Record Trace - IFCID 402 - System Profile - Monitoring Statistics (DB2 10 or later)”.

IFCID 402 records information about any profile warnings or exception conditions that occurred during a Statistics interval. Each trace record can contain information for up to 500 unique profiles. Multiple trace records are written if profile thresholds are exceeded for more than 500 unique profiles during a given Statistics interval.

This record is written when Statistics class 4 is on.

Record trace - IFCID 402 - System Profile - Monitoring Statistics (DB2 10 or later)

The field labels shown in the following sample layout of “Record Trace - IFCID 402 - System Profile - Monitoring Statistics (DB2 10 or later)” are described in the following section.

PROFILE ID

The profile ID.

Field Name: QW0402PI

THR EXC TSH EXCEEDED

The accumulated counter of thread exception threshold that was exceeded.

Field Name: QW0402TE

THR QUEUED/SUSP WHEN EXC TSH WAS EXCEEDED

The accumulated counter of the thread that was queued or suspended when the thread exception threshold was exceeded.

Field Name: QW0402TQ

REQUEST FAILED WHEN THR EXC TSH WAS EXCEEDED

The accumulated counter of the request that failed when the thread exception threshold was exceeded.

Field Name: QW0402TF

THR WARNING TSH BEING EXCEEDED

The accumulated counter of thread warning threshold that was exceeded.

Field Name: QW0402TW

CONNECTION EXC TSH BEING EXCEEDED

Accumulated counter of the connection exception threshold that was exceeded.
**Field Name: QW0402CE**

**CONNECTION WARN TSH BEING EXCEEDED**

The accumulated counter of the connection warning threshold that was exceeded.

**Field Name: QW0402CW**

**IDLE THR EXC TSH BEING EXCEEDED**

The accumulated counter of the idle thread exception threshold that was exceeded.

**Field Name: QW0402OE**

**IDLE THR WARN TSH BEING EXCEEDED**

The accumulated counter of the idle thread warning threshold that was exceeded.

**Field Name: QW0402OW**
IFCID 497 - Non Nested Statement ID Record

This topic shows detailed information about “Record Trace - IFCID 497 - Non Nested Statement ID Record”.

IFCID 497 (Non Nested Statement ID Record), IFCID 498 (UDF Statement ID Record), and IFCID 499 (Stored Procedure Statement ID Record) have the same mapping structure comprising two data sections.

Record trace - IFCID 497 - Non Nested Statement ID Record

The field labels shown in the following sample layout of “Record Trace - IFCID 497 - Non Nested Statement ID Record” are described in the following section.

REASON IFCID WAS EXTERNALIZED

Identifies the reason why this IFCID was externalized.

Field Name: QW0499RS

STATEMENT ID

The unique statement identifier. It is shown as an integer and in hexadecimal format.

Field Name: QW0499SID

CONV INTO HEX

The unique statement identifier. It is shown as an integer and in hexadecimal format.

Field Name: QW0499SID

EXECUTIONS

The number of executions.

Field Name: QW0499NEC

TYPE

The statement type. Possible values are DYNAMIC or STATIC. In addition, the statement type of IFCID 499 can also have the values STATIC CALL or DYNAMIC CALL.

Field Name: QW0499STY
IFCID 498 - UDF Statement ID Record

This topic shows detailed information about “Record Trace - IFCID 498 - UDF Statement ID Record”.

IFCID 497 (Non Nested Statement ID Record), IFCID 498 (UDF Statement ID Record), and IFCID 499 (Stored Procedure Statement ID Record) have the same mapping structure comprising two data sections.

Record trace - IFCID 498 - UDF Statement ID Record

The field labels shown in the following sample layout of “Record Trace - IFCID 498 - UDF Statement ID Record” are described in the following section.

<table>
<thead>
<tr>
<th>Reason IFCID was externalized: A UDF is ending</th>
</tr>
</thead>
<tbody>
<tr>
<td>STATEMENT ID : 332108 CONV INTO HEX: X'000000000005114C' EXECUTIONS : 1</td>
</tr>
<tr>
<td>TYPE : STATIC</td>
</tr>
<tr>
<td>STATEMENT ID : 332109 CONV INTO HEX: X'000000000005114D' EXECUTIONS : 1</td>
</tr>
<tr>
<td>TYPE : STATIC</td>
</tr>
</tbody>
</table>

REASON IFCID WAS EXTERNALIZED

Identifies the reason why this IFCID was externalized.

Field Name: QW0499RS

STATEMENT ID

The unique statement identifier. It is shown as an integer and in hexadecimal format.

Field Name: QW0499SID

CONV INTO HEX

The unique statement identifier. It is shown as an integer and in hexadecimal format.

Field Name: QW0499SID

EXECUTIONS

The number of executions.

Field Name: QW0499NEC

TYPE

The statement type. Possible values are DYNAMIC or STATIC. In addition, the statement type of IFCID 499 can also have the values STATIC CALL or DYNAMIC CALL.

Field Name: QW0499STY
IFCID 499 - Stored Procedure Statement ID Record

This topic shows detailed information about “Record Trace - IFCID 499 - Stored Procedure Statement ID Record”.

IFCID 497 (Non Nested Statement ID Record), IFCID 498 (UDF Statement ID Record), and IFCID 499 (Stored Procedure Statement ID Record) have the same mapping structure comprising two data sections.

Record trace - IFCID 499 - Stored Procedure Statement ID Record

The field labels shown in the following sample layout of “Record Trace - IFCID 499 - Stored Procedure Statement ID Record” are described in the following section.

REASON IFCID WAS EXTERNALIZED
Identifies the reason why this IFCID was externalized.

Field Name: QW0499RS

STATEMENT ID
The unique statement identifier. It is shown as an integer and in hexadecimal format.

Field Name: QW0499SID

CONV INTO HEX
The unique statement identifier. It is shown as an integer and in hexadecimal format.

Field Name: QW0499SID

EXECUTIONS
The number of executions.

Field Name: QW0499NEC

TYPE
The statement type. Possible values are DYNAMIC or STATIC. In addition, the statement type of IFCID 499 can also have the values STATIC CALL or DYNAMIC CALL.

Field Name: QW0499STY
Chapter 41. The Record Trace File Data Set and Output Records

The record trace file data set is a sequential data set of formatted records suitable for loading into the performance database using the DB2 load utility and from which reports can be produced using a reporting facility such as Query Management Facility (QMF).

The output of the FILE subcommand is a sequential variable blocked data set. The content of the output data set is determined by the FILE command options you specify, and by the input SMF/GTF records processed.

This data set contains SQL statement records and RID pool records.

Descriptions of the Record Trace File data sets and the fields contained can be found in the RKO2SAMP library under the following names:

**DGONDFMB**
For Minibind (IFCID 022) records

**DGONDFSQ**
For SQL Statement (IFCID 063) records

**DGONDFSE**
For Sort End (IFCID 096) records

**DGONDFRP**
For RID Pool (IFCID 125) records
Part 8. The SQL Activity Report Set

These topics provide information about the SQL activity reports.

Note:
1. Refer also to the sections of Reporting User’s Guide that deal with SQL Activity.
2. For an introduction to the SQL Activity report set and general SQL Activity information refer to the Reporting User’s Guide

Chapter 42, “Introduction to the SQL Activity Report Set,” on page 42-1
The SQL Activity report set consists of reports and traces and provides information on the SQL activity taking place during the processing of a DB2 application. The reports show the processing of an SQL statement and all the related DB2 activity, known as workload, related to that statement.

Chapter 43, “General SQL Activity Information,” on page 43-1
Here you find information common to all SQL Activity reports and traces.

Chapter 44, “The SQL Activity Report,” on page 44-1
This topic shows examples of SQL Activity reports and the commands used to generate them.

Chapter 45, “The SQL Activity Trace,” on page 45-1
This topic contains examples of SQL Activity traces and the commands used to generate them.

Chapter 46, “SQL Activity Report and Trace Blocks,” on page 46-1
Here you find the blocks reported by SQL Activity report and trace.
Chapter 42. Introduction to the SQL Activity Report Set

The SQL Activity report set consists of reports and traces and provides information on the SQL activity taking place during the processing of a DB2 application. The reports show the processing of an SQL statement and all the related DB2 activity, known as workload, related to that statement.

Also included in these reports are trigger and nesting level information. When the appropriate IFCID 003 is present, Accounting Trace is also reported.

In most situations, an SQL Activity trace gives details on either a DB2 thread or part of a reused thread between two signons. In CP query and sysplex query parallelism, an originating thread and multiple parallel threads are created to execute an SQL statement. Data from parallel threads can also be included in the originating thread if you specify an INPUTDD containing the relevant data in your JCL. The term thread is used to include the originating and the parallel threads. The trace is a collection of threads presented in logical unit of work ID (LUWID) sequence with an index to help you find a particular thread.

An SQL Activity report is an aggregation of threads ordered by the combination of OMEGAMON XE for DB2 PE identifiers you specify. If you specify no OMEGAMON XE for DB2 PE identifiers with ORDER, the default order of PRIMAUTH and PLANNAME is used. The report is a summary of all the work belonging to, and ordered by, those identifiers.

The following information is collected for each thread provided that the appropriate IFCIDs are available:

- Thread identification (OMEGAMON XE for DB2 PE identifiers, DB2 logical unit of work ID (LUWID), CICS logical unit of work ID (LUWID), ACE, thread start and stop time, thread type, and location)
- Programs (DBRMs and packages), stored procedures, cursors used, UDF, and triggers.
- SQL statements executed within the thread with their workload detail
- Events and time spent in DDF processing
- Time spent in signon processing
- Time spent creating and terminating threads
- Time spent in autobind processing
- Accounting information

You can control the level of summarization, the sorting of events, and the workload detail within a unit of reporting.

Explanation of long names

DB2 supports long values, up to 128 characters, for certain fields. When this happens, the value shown by OMEGAMON XE for DB2 PE in the report block can be truncated, depending on the space available. Truncated values are then listed at the end of each logical report unit, together with the full values.
When a value is truncated, the “greater than” sign (>) is printed instead of a colon (:) following the label name. The full value starts with a "greater than" sign followed by the label. For example:

\n\nTname > This value
...  
...  
>Tname : This value was truncated
...  
\nIf truncated values are listed, the "greater than" sign (>) is shown at the end of each value, because there is no colon (:) as a delimiter between the label and the value. In lists the label is used as a column heading.

For example:

<table>
<thead>
<tr>
<th>ACCESS_CREATOR</th>
<th>ACCESS_NAME</th>
<th>MATCHCOLS</th>
<th>INDEXONLY</th>
<th>PREFETCH_INDEX</th>
<th>OPERATION</th>
<th>MIXOPSEQ</th>
</tr>
</thead>
<tbody>
<tr>
<td>TDK_LONG&gt;</td>
<td>IX_OMPE_FIRST_LONG&gt;</td>
<td>0</td>
<td>YES</td>
<td>SEQUENTIAL</td>
<td>SCAN</td>
<td>1</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&gt;ACCESS_CREATOR</td>
<td>TDK_LONG_NAMED_COLLECTION_FOR_LONG_NAMED_OBJECTS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&gt;ACCESS_NAME</td>
<td>IX_OMPE_FIRST_LONG_NAMED_TABLE_FOR_UNCOMMITTED_READ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The mapping between truncated and full values remains the same for multiple reports from the same input data. This mapping for multiple reports from different input data cannot be guaranteed.
Chapter 43. General SQL Activity Information

Here you find information common to all SQL Activity reports and traces.

Summarization” on page 43-2
When you summarize statements, statements with similar characteristics are merged into a single record. As an example, if you were to summarize by statement number and you had 3 occurrences of statement 777, one of 778 and four of 779 in the same package, one for each statement number would be created.

“Sorting” on page 43-4
This topic explains how to sort events in a summarized report or trace.

“Workload Detail” on page 43-5
The workload figures are applied to the event being summarized.

“Headers Used in SQL Activity” on page 43-6
A header is printed at the top of every SQL Activity trace, report, and index page.
Summarization

When you summarize statements, statements with similar characteristics are merged into a single record. As an example, if you were to summarize by statement number and you had 3 occurrences of statement 777, one of 778 and four of 779 in the same package, one for each statement number would be created.

You can summarize the SQL events by any combination of:

**Cursor**

Statements with the same cursor name in the same DBRM or package name are grouped into one record.

**Occurrence**

All SQL statement occurrences are shown as separate records. Statements are not merged. This is valid for traces only, and is the default. When summarization by occurrence is shown, the nesting level of the trigger or user-defined function is also shown.

**Program**

Statements with the same DBRM or package name are grouped into one record. This is the default for reports.

**Statement number**

Statements with the same statement number in the same DBRM or package name are grouped into one record.

**Statement type**

Statements of the same statement type are grouped into one record.

SQL Activity trace automatically includes a summary by thread and, if there is more than one thread per location, a summary by location.

**Note:**

1. The statement number is printed in a summary by cursor if an SQL statement does not include the cursor name, or if summarization by cursor is not appropriate for that statement. The following SQL statements are not summarized by cursor:
   - `CALL`
   - `DELETE` (noncursor or noncurrent of cursor)
   - `DESCRIBE`
   - `INSERT`
   - `PREPARE`
   - `SQL COMMIT`
   - `SQL ROLLBACK`
   - `SQL statement at application requester using DRDA`
   - `UPDATE` (noncursor or noncurrent of cursor).
   In a summary by cursor, these statements are organized by the statement number.

2. The statement type is printed in a summary by cursor and a summary by statement number if an SQL statement does not include the statement number or cursor name.

3. If DDL and DCL statements are present in a summary by cursor or a summary by statement number, they are organized by statement type.
SQL Activity prints all the SQL it receives. If some SQL cannot be summarized at the requested level, it is presented in the closest possible summary format.
This topic explains how to sort events in a summarized report or trace.

They can be sorted by:
- Average elapsed time
- Average TCB time
- Default
- Exits
- Exit time
- I/O requests
- I/O time
- Lock suspensions
- Lock suspension time
- Number of scans
- Pages scanned
- Records sorted
- Rows processed
- Sort workfiles

If DEFAULT is used, the sort order is dictated by the summarization, as follows:

**Cursor**
Sorted in alphabetical order within program

**Occurrence**
Sorted in timestamp order (trace default)

**Program**
Sorted in alphabetical order of package or DBRM name (report default)

**Statement number**
Sorted in numerical order within program

**Statement type**
Sorted in alphabetical order.
Workload Detail

The workload figures are applied to the event being summarized.

Any combination of the following workload detail can be requested:
- Accounting, see 2
- All
- Data capture activity
- Exit activity
- I/O activity
- Locking activity
- Minibind, see 3
- None
- Scan, RID list, and query parallelism activity
- Sort activity
- UDF, see 4
- Vars
- Workload highlights

None is the default.

Note:
1. The amount of processing required depend on the level of detail you request. IBM recommends that you do not specify WORKLOAD(ALL) with a large amount of input data unless absolutely necessary.
2. When IFCID 003 is included in the input, Accounting Trace activity is automatically included as part of the workload detail.
3. When IFCID 022 is included in the input, minibind activity is included automatically as part of the workload detail.
4. When IFCID 324 is included in the input, UDF activity is automatically included as part of the workload detail.
Headers Used in SQL Activity

A header is printed at the top of every SQL Activity trace, report, and index page.

The following topics provide additional information:

- “SQL Activity Report Header” on page 43-8
- “SQL Activity Trace Header” on page 43-9
- “SQL Activity Trace Index Header” on page 43-10

The report headers contain the standard OMEGAMON XE for DB2 PE header information and the following additional data:

SORTED BY
The event by which the report or trace is sorted, which can be the default or as specified in the SORTBY option.

WITH detail WORKLOAD
The workload details included in the report or trace as specified in the WORKLOAD option.

OMEGAMON XE for DB2 PE identifiers
The identifiers define the order of the SQL Activity data printed. If you specify no OMEGAMON XE for DB2 PE identifiers with ORDER, the default order of PRIMAUTH-PLANNAME is used.

TRACE #
Each trace occurrence and thread within the TRACE subcommand is numbered sequentially in the format x.yyyyy, where:
- x can be 1 through 5, representing the five TRACE subcommands
- yyyy can be 1 through 99 999, representing each thread being traced.

DB2 LUWID
The identifier of the logical unit of work. The following parts of this identifier are printed:
- The network ID
- The name of the logical unit, which is the name by which VTAM recognizes the DB2 subsystem
- The instance number

ACE ADDRESS
The agent control element absolute address in hexadecimal.

START TIME
The timestamp showing when the startup of the thread ended or, if the REQUESTED FROM time is after the thread begin, the REQUESTED FROM timestamp.

START ELAPSED
The thread start elapsed time, if calculable.

START REASON
The event that started the thread:
- CREATE THREAD
- CREATE DBAT
- NEW USER
- RESIGNON
- IN PROGRESS, no thread start IFCID present.
STOP TIME
The timestamp showing when the thread stopped or, if the REQUESTED TO time is after the thread end, the REQUESTED TO timestamp.

STOP ELAPSED
The thread stop elapsed time, if calculable.

STOP REASON
The event that stopped the thread:
• TERMINATE THREAD
• DEALLOCATE DBAT
• NEW USER
• RESIGNON
• ACCOUNTING FOUND-the thread terminated with the accounting record
• LOCATION CHANGED-the thread was terminated due to a location change
• END OF FILE-the thread was terminated because there were no records left to process.

START AET
The average thread start elapsed time.

STOP AET
The average thread stop elapsed time.

AUTOBIND AET
The average autobind elapsed time, if present.

ARCHIVE LOG AET
The average archive log (quiesce) elapsed time, if present.

This is an example of the SQL Activity report header.
This section introduces the header of the SQL Activity trace.
This is an example of the SQL Activity Trace Index Header.
### SQL Activity Report Header

This is an example of the SQL Activity report header.

<table>
<thead>
<tr>
<th>EVENT</th>
<th>COUNT</th>
<th>TOTAL TCB</th>
<th>AET/EVENT</th>
<th>TCB/EVENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSNESM68</td>
<td>1</td>
<td>0.134154</td>
<td>0.007978</td>
<td></td>
</tr>
</tbody>
</table>

SQL ACTIVITY REPORT COMPLETE
SQL Activity Trace Header

This section introduces the header of the SQL Activity trace.

SQL Activity Trace Header

Here is an example of an SQL Activity trace header.

LOCATION: DSNAPC3  
GROUP: GROUP_1  
MEMBER: MEMBER_1  
SUBSYSTEM: APC3  
DB2 VERSION: V10

SUMMARIZED BY OCCURRENCE

PRIMAUTH: XXASP09  
ORIGAUTH: XXASP09  
ENDUSER: 1234567890123456  
CONNTYPE: TSO

CONNECT: BATCH  
PLANNAME: LOCURHIL  
TRANSACTION: 12345678901234567890123456789012

CORRNAME: XXASP09F  
CORRNMFR: 'BLANK'  
TRANSACT: 12345678901234567890123456789012

TO: NOT SPECIFIED

ACTUAL FROM: 01/30/10 03:28:52.13
Chapter 44. The SQL Activity Report

This topic shows examples of SQL Activity reports and the commands used to generate them.

These example reports are summarized and sorted at the following levels:
  • Statement number
  • Cursor
  • Program
  • Statement type
  • All

The SQL Activity report groups SQL Activity according to a combination of up to three OMEGAMON XE for DB2 PE identifiers. This grouping is applied to any SUMMARIZEBY, SORTBY, or WORKLOAD options you specify.

The ORDER subcommand specifies by which OMEGAMON XE for DB2 PE identifiers, and in which order, the SQL Activity is reported. If you specify no OMEGAMON XE for DB2 PE identifiers with ORDER, the default order of PRIMAUTH and PLANNAME is used.

"Examples of an SQL Activity Report" on page 44-2
This topic shows examples of an SQL Activity report.

"Example of an SQL Activity Report with Workload" on page 44-5
The following examples show excerpts of an SQL Activity report with workload detail.
Examples of an SQL Activity Report

This topic shows examples of an SQL Activity report.

“Summarized by Statement Number” to “Summarized by Statement Type” on page 44-3 show excerpts of an SQL Activity report summarized by all. The layout of the report is similar for each of the possible summary levels. The order is plan name within primary authorization ID, by default.

Summarized by Statement Number

This summary level presents totals for each statement number belonging to the selected combination of OMEGAMON XE for DB2 PE identifiers. The events are qualified by package name.

By default, the package names are sorted alphabetically and the statement numbers within packages are sorted numerically, in ascending order.

Note: Not every statement can be summarized by statement number. DDL, for example, has no statement numbers. An event name is chosen from the closest possible level of summarization, which is the statement type LOCK in this example.

The following command produces an SQL Activity report summarized by Statement Number:

```
... SQLACTIVITY REPORT SUMMARIZEBY (STMTNO) ...
```

This is an example for an SQL Activity report summarized by statement number.

```
LOCATION: SYS1DSN2 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-1
GROUP: DSN2 SQL ACTIVITY - REPORT REQUESTED FROM: NOT SPECIFIED
MEMBER: SE11 TO: NOT SPECIFIED
SUBSYSTEM: SE11 ORDER: PRIMAUTH-PLANNAME ACTUAL FROM: 01/30/10 06:55:37.58 DB2 VERSION: V10 TO: 01/30/10 07:05:37.61

SUMMARIZED BY STMTNO
PRIMAUTH: WRL PLANNAME: PARALCPU THREAD TOTAL: 11 START AET: 0.053771 STOP AET: N/P

EVENT COUNT TOT.ELAPS TOTAL TCB DETAIL
-------------------- ----------- ----------- ---------
PACKAGE SYS1DSN2.PARAL.PARALC01.X'158A622D10FD8B50' DB2OMPETEST
ACQUIRE(USE) REOPT(N) RELEASE(COMMIT) ISO(CS) DYNAMICRULES(RUN)
PREPARE(NODEFER) KEEPDYNAMIC(NO) PROTOCOL(DRDA) OPTHINT(N/P)
IMMEDWRITE(PH1)

# 120 1 7:10.524819 47.134431 OPEN CURSOR: CRS1 ISO(CS) REOPT(NO) KEEP UPD LOCKS: NO
7:10.524819 47.134431
# 137 12888 36.562407 8.188774 FETCH CURSOR: CRS1
0.002837 0.000635
```

Summarized by Cursor

This summary level shows totals for each cursor name belonging to the selected combination of OMEGAMON XE for DB2 PE identifiers that are qualified by package name. By default, the package names and the events within each package are sorted alphabetically.

This is an example for an SQL Activity report summarized by cursor.

```
LOCATION: SYS1DSN2 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-2
GROUP: DSN2 SQL ACTIVITY - REPORT REQUESTED FROM: NOT SPECIFIED
MEMBER: SE11 TO: NOT SPECIFIED

SUMMARIZED BY STMTNO
PRIMAUTH: WRL PLANNAME: PARALCPU THREAD TOTAL: 11 START AET: 0.053771 STOP AET: N/P

DETAIL

PACKAGE SYS1DSN2.PARAL.PARALC01.X'15BA622D10FD8B50' DB2OMPETEST
ACQUIRE(USE) REOPT(N) RELEASE(COMMIT) ISO(CS) DYNAMICRULES(RUN)
PREPARE(NODEFER) KEEPDYNAMIC(NO) PROTOCOL(DRDA) OPTHINT(N/P)
IMMEDWRITE(PH1)

# 120 1 7:10.524819 47.134431 OPEN CURSOR: CRS1 ISO(CS) REOPT(NO) KEEP UPD LOCKS: NO
7:10.524819 47.134431
# 137 12888 36.562407 8.188774 FETCH CURSOR: CRS1
0.002837 0.000635
```
This summary level presents totals for all programs belonging to the selected combination of OMEGAMON XE for DB2 PE identifiers. Package names are embedded in the summary details.

This is an example for an SQL Activity report summarized by program.

This summary level shows totals for each cursor name belonging to the selected combination of OMEGAMON XE for DB2 PE identifiers that are qualified by package name. By default, the package names and the events within each package are sorted alphabetically.

This is an example for an SQL Activity report summarized by statement type.
**Summarized by Thread**

This summary level presents totals for each statement type executed by this combination of OMEGAMON XE for DB2 PE identifiers. By default, the events are sorted alphabetically. There is no further qualification at this level.

This is an example for an SQL Activity report summarized by thread.

<table>
<thead>
<tr>
<th>LOCATION: OMPDB2L</th>
<th>GROUP: DB2L</th>
<th>MEMBERS: SDL2</th>
<th>SUBSYSTEM: SDL2</th>
<th>DB2 VERSION: V10</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRIMAUTH: N/P</td>
<td>CONNECT: N/P</td>
<td>CORRNAME: N/P</td>
<td>CONNTYPE: 'BLANK'</td>
<td></td>
</tr>
<tr>
<td>ORIGAUTH: N/P</td>
<td>PLANNAME: N/P</td>
<td>CORRMBR:</td>
<td>THRTYPE: ALLIED</td>
<td></td>
</tr>
<tr>
<td>ENQUSER: N/P</td>
<td>WSNAME: N/P</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ENDUSER: N/P</td>
<td>WSNAME: N/P</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TRACE # 1.1</td>
<td>DB2 UDID: DEIBMIPS.IPUAWDL2.X'C4B7EAC08924'</td>
<td>ACE ADDRESS: X'1AC558F8'</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**SUMMARIZED BY THREAD**

<table>
<thead>
<tr>
<th>EVENT</th>
<th>COUNT</th>
<th>TOT.ELAPS</th>
<th>TOTAL TCB</th>
<th>DETAIL</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>AET/EVENT</td>
<td>TCB/EVENT</td>
<td></td>
</tr>
<tr>
<td>X'C4B7EAC08924'</td>
<td>1</td>
<td>6.823537</td>
<td>0.025761</td>
<td>CREATE</td>
</tr>
</tbody>
</table>

**LOCATION:** OMPDB2L
**GROUP:** DB2L
**MEMBER:** SDL2
**SUBSYSTEM:** SDL2
**DB2 VERSION:** V10

**SUMMARIZED BY THREAD**

<table>
<thead>
<tr>
<th>EVENT</th>
<th>COUNT</th>
<th>TOT.ELAPS</th>
<th>TOTAL TCB</th>
<th>DETAIL</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>AET/EVENT</td>
<td>TCB/EVENT</td>
<td></td>
</tr>
<tr>
<td>X'C4B7EAC08924'</td>
<td>1</td>
<td>6.823537</td>
<td>0.025761</td>
<td>CREATE</td>
</tr>
</tbody>
</table>
Example of an SQL Activity Report with Workload

The following examples show excerpts of an SQL Activity report with workload detail.

They are generated by the following command:

```
SQLACTIVITY
REPORT
SUMMARIZEBY (ALL)
WORKLOAD (ALL)
```

Summarized by Statement Number, with All Workload

This page of the report shows the summary by statement number for primary authorization ID WRL and plan name DSNESPRR for location SYS1DSN2. It includes all workload contained in the input.
Summarized by Cursor, With All Workload

This page of the report shows the summary by cursor for primary authorization ID WRL and plan name DSNESPRR for location SYS1DSN2. It includes all workload contained in the input.

LOCATION: SYS1DSN2
GROUP: DSN2
MEMBER: SE21
SUBSYSTEM: DSN2
DB2 VERSION: V10
SUMMARIZED BY CURSOR, WITH ALL WORKLOAD

--- EVENT ---

<table>
<thead>
<tr>
<th>THREAD</th>
<th>ACTUAL FROM</th>
<th>STOP AET:</th>
<th>START AET:</th>
<th>ACTUAL FROM</th>
<th>STOP AET:</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>01/30/10 06:55:37.58</td>
<td>N/P</td>
<td>N/P</td>
<td>01/30/10 07:05:37.61</td>
<td>N/P</td>
</tr>
</tbody>
</table>

--- DBRM ---

<table>
<thead>
<tr>
<th>MEMBER</th>
<th>TYPE</th>
<th>DBRM</th>
<th>COUNT</th>
<th>TOT.ELAPS</th>
<th>TOTAL TCB</th>
<th>THREAD TOTAL</th>
<th>DETAIL</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>DSNESM68</td>
<td>2</td>
<td>2:15.220670</td>
<td>0.785512</td>
<td>11</td>
<td></td>
</tr>
</tbody>
</table>

--- SCAN ACTIVITY ---

--- PAGE & ROW LOCKING ---

--- LOCK SUSPENSION ACTIVITY ---

--- WORKLOAD HILITE ---

--- WORKLOAD HILITE ---

Summarized by Program, With All Workload

This page of the report shows the summary by program for primary authorization ID WRL and plan name DSNESPRR for location SYS1DSN2. It includes all workload contained in the input.

LOCATION: SYS1DSN2
GROUP: DSN2
MEMBER: SE21
SUBSYSTEM: DSN2
DB2 VERSION: V10
SUMMARIZED BY PROGRAM, WITH ALL WORKLOAD

--- EVENT ---

--- DBRM ---

--- SCAN ACTIVITY ---

--- PAGE & ROW LOCKING ---

--- LOCK SUSPENSION ACTIVITY ---

--- WORKLOAD HILITE ---

--- WORKLOAD HILITE ---
Summarized by Statement Type, with All Workload

This page of the report shows the summary by statement type for primary authorization ID WRL and plan name DSNESPRR for location SYS1DSN2. It includes all workload contained in the input.

LOCATION: SYS1DSN2
GROUP: DSN2
MEMBER: SE21
SUBSYSTEM: SE21
DB2 VERSION: V10

PRIMAUTH: WRL PLANNAME: DSNESPRR

--- WORKLOAD HILITE
SCANS : 8 RECS/SORT: 3.00 I/O REQS: 1 SUSPENDS : 2 EXITS : 2 AMS : 1
RONSPROC: 8 WORK/SORT: 2.00 AET/I/O : 1.374752 AET/SUSP : 0.485483 AET/EXIT : 0.048234 AET/AMS : 0.094745
PAGESCAN: 47 PASS/SORT: 2.00 DATACAPT: YES KIDS UNUSED: 2 CHECKCON : REJECTED DEGREE REDUCTION : 3
LOB_PAGSCAN: 12345 LOB_UPD_PAGE: 12345

--- SCAN ACTIVITY
DATABASE MEMBER TYPE
DSNDB04 DGO71J5L 12 53 53 330220 4 4 00
SE21 SEQD
DSNDB04 DGO719AX 12 43 43 2 50200 4 4 00
SE21 SEQD
DSNDB06 SYSDBASE 70 70 0 7 00000 1 6 00
SE21 INDEX
DSNDB06 SYSDBASE 46 46 0 600000 4 6 00
SE21 SEQD
TOTAL 140 212 142 12 80420 2 9 8 00

--- LOCK SUSPENSION ACTIVITY
RESOURCE NAME MEMBER TYPE REQUEST LOCAL LATCH IRLMQ GROUP NOTIF OTHER COUNT AET COUNT AET COUNT AET
N/P N/P CHANGE 0 0 0 1 0 0 1 0.09668 0 N/C 0 N/C
SE21 N/P LOCK 0 0 0 2 0 0 2 0.14444 0 N/C 0 N/C
DSNDB04 DGO71J5L PAGESET LOCK 0 0 0 2 0 0 2 0.14444 0 N/C 0 N/C
DSNDB04 DGO719AX P/P CAST LOCK 0 0 0 1 0 0 1 0.08707 0 N/C 0 N/C

TOTAL 6 0

Chapter 44. The SQL Activity Report
<table>
<thead>
<tr>
<th>DATABASE</th>
<th>PAGESET</th>
<th>SCANS</th>
<th>PROCESS</th>
<th>EXAMINE</th>
<th>STAGE 1</th>
<th>STAGE 2</th>
<th>INSERTS</th>
<th>UPDATES</th>
<th>DELETES</th>
<th>SCANNED</th>
<th>SCANS</th>
<th>DELETES</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSN0804</td>
<td>DG071J5L</td>
<td>6</td>
<td>31</td>
<td>31</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>24</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SE21</td>
<td>SE00</td>
<td>6</td>
<td>25</td>
<td>25</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>24</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TOTAL</td>
<td></td>
<td>12</td>
<td>56</td>
<td>56</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>48</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

--- LOCK SUSPENSION ACTIVITY

<table>
<thead>
<tr>
<th>RESOURCE NAME</th>
<th>TYPE</th>
<th>REQUEST</th>
<th>LOCAL</th>
<th>LATCH</th>
<th>IRLMQ</th>
<th>GROUP</th>
<th>NOTIF</th>
<th>OTHER</th>
<th>COUNT</th>
<th>AET</th>
<th>COUNT</th>
<th>AET</th>
<th>COUNT</th>
<th>AET</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSN0804</td>
<td>DG071J5L</td>
<td>PAGESET</td>
<td>LOCK</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>59.5816</td>
<td>1</td>
<td>72.4844</td>
<td>0</td>
<td>N/C</td>
</tr>
<tr>
<td>INSERT</td>
<td>4</td>
<td>2.092164</td>
<td>0.032354</td>
<td>0.523041</td>
<td>0.008089</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Chapter 45. The SQL Activity Trace

This topic contains examples of SQL Activity traces and the commands used to generate them.

These traces are summarized at different levels. They can be summarized at the following levels:
- Occurrence
- Statement number
- Cursor
- Program
- Statement type

Note: The trace also automatically includes a summary by thread and, if there is more than one thread per location, a summary by location. Each summary begins on a new page of the trace.

"Example of an SQL Activity Trace" on page 45-2
The following examples show excerpts of an SQL Activity trace summarized at all levels.

"The SQL Activity Trace Index" on page 45-7
The SQL Activity trace index provides a page index to the threads traced during the execution of an SQLACTIVITY command. An SQL Activity trace index is produced for each TRACE subcommand and is printed on a new page at the end of the trace output.
Example of an SQL Activity Trace

The following examples show excerpts of an SQL Activity trace summarized at all levels.

They are generated by the following command:

```
>: SQLACTIVITY
  TRACE
    SUMMARIZEY (ALL)
```

SQL Activity Trace Summarized by Occurrence

Summary by occurrence shows individual SQL statement occurrences. In this thread the SQL statements belong to one package, the name of which is printed at the head of its work. When present, SQL text and DDF information is embedded in the events. Commits appear as standalone events. The events are, by default, sorted in timestamp sequence.

This is an example of an SQL Activity Trace Summarized by Occurrence.
SQL Activity Trace Summarized by Occurrence with All Workload

Summary by occurrence with all workload shows individual SQL statement occurrences. It also shows workload highlights, scan activity and minibind activity.

This is an example of an SQL Activity Trace Summarized by Occurrence with All Workload.

Chapter 45. The SQL Activity Trace 45-3
SQL Activity Trace Summarized by Statement Number

This summary level presents totals for each statement number belonging to the thread. The events are qualified by the package name.

From this level on, timestamps are not appropriate so the second column becomes a count of the occurrences of each event. SQL text is omitted.

By default, the package names are printed alphabetically and the statement numbers are sorted numerically within each package.

Note: Not every statement can be summarized by statement number. DCL, for example, has no statement number. An event name is chosen from the closest possible level of summarization, which is the statement type GRANT in this example.

This is an example of an SQL Activity Trace Summarized by Occurrence with All Workload.

SQL Activity Trace Summarized by All-Cursor

This summary level shows totals for each cursor name belonging to the thread that is qualified by package name. By default, the package names and the events within each package are sorted alphabetically.

This is an example of an SQL Activity Trace Summarized by All-Cursor.
SQL Activity Trace Summarized by Program

This summary level presents totals for each program name belonging to the thread.

SQL Activity Trace Summarized by Statement Type

This summary level presents totals for each statement type executed by the thread. By default, the events are sorted alphabetically. There is no further qualification at this level.
SQL Activity Trace Summarized by Thread

These totals are printed to further summarize the SQL in this thread.

This is an example of an SQL Activity Trace Summarized by Statement Type.
The SQL Activity Trace Index

The SQL Activity trace index provides a page index to the threads traced during the execution of an SQLACTIVITY command. An SQL Activity trace index is produced for each TRACE subcommand and is printed on a new page at the end of the trace output.

Each thread on the trace is listed in timestamp order followed by various OMEGAMON XE for DB2 PE identifiers.

If a thread satisfies the selection criteria for more than one TRACE subcommand, it is presented on each relevant trace, and is indexed accordingly.

SQL Activity Trace Index

The following example represents the trace index for location SYS1DSN2. From this page you can see the following:

- This trace is the result of the first TRACE subcommand.
- There are four threads.
- There were no dates or times specified on the TRACE subcommand or on the GLOBAL command.
- Both allied threads and DBATs are present. For allied threads, the requester location is the same as the local location (for example, the first entry shown in “SQL Activity Trace Index”). For DBATs, the requester location is different to the local location (for example, the second entry shown in “SQL Activity Trace Index”).

<table>
<thead>
<tr>
<th>LOCATION: SYS1DSN2</th>
<th>OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0)</th>
<th>PAGE: 0-2</th>
</tr>
</thead>
<tbody>
<tr>
<td>GROUP: DDN2</td>
<td>SQL ACTIVITY - TRACE</td>
<td>REQUESTED FROM: NOT SPECIFIED</td>
</tr>
<tr>
<td>MEMBER: SE21</td>
<td></td>
<td>TO: NOT SPECIFIED</td>
</tr>
<tr>
<td>DB2 VERSION: V10</td>
<td></td>
<td>ACTUAL FROM: 01/30/10 06:45:39.34</td>
</tr>
</tbody>
</table>

SQL Trace # 1

<table>
<thead>
<tr>
<th>TRACE START</th>
<th>CONNECT</th>
<th>CORNAME</th>
<th>CORRNAME</th>
<th>PRIMAUTH</th>
<th>ORIGAUTH</th>
<th>PLANNAME</th>
<th>REQUESTER</th>
<th>INSTANCE</th>
<th>ACE ADDRESS</th>
<th>TRACE NO</th>
<th>PAGE NO</th>
</tr>
</thead>
<tbody>
<tr>
<td>01/30/10 06:42:18.13 TSO</td>
<td>WRL</td>
<td>‘BLANK’</td>
<td>WRL1</td>
<td>WRL1</td>
<td>DSNESPRR</td>
<td>SYS1DSN2</td>
<td>A07F37CCE027</td>
<td>0540F38BB</td>
<td>1.7</td>
<td>2-69</td>
<td></td>
</tr>
<tr>
<td>01/30/10 06:45:39.34 TSO</td>
<td>WRL</td>
<td>‘BLANK’</td>
<td>WRL</td>
<td>WRL</td>
<td>DSNESPRR</td>
<td>PMOD2E21</td>
<td>A07F37C1D020</td>
<td>05CDA8C8</td>
<td>1.4</td>
<td>2-1</td>
<td></td>
</tr>
<tr>
<td>01/30/10 06:49:32.14 TSO</td>
<td>WRL</td>
<td>‘BLANK’</td>
<td>WRL</td>
<td>WRL</td>
<td>DSNESPRR</td>
<td>PMOD2E21</td>
<td>A07F383B0360</td>
<td>05CDA8C8</td>
<td>1.5</td>
<td>2-7</td>
<td></td>
</tr>
<tr>
<td>01/30/10 06:55:44.54 TSO</td>
<td>WRL</td>
<td>‘BLANK’</td>
<td>WRL</td>
<td>WRL</td>
<td>DSNESPRR</td>
<td>PMOD2E21</td>
<td>A07F3A035F9</td>
<td>05CDA8C8</td>
<td>1.6</td>
<td>2-38</td>
<td></td>
</tr>
</tbody>
</table>

SQL ACTIVITY TRACE COMPLETE

Field description

Here is a description of the field labels shown in the trace summary index:

TRACE START

The start time of the trace of the thread.

OMEGAMON XE for DB2 PE identifiers

The identifiers define the order of the SQL Activity data printed. If you specify no OMEGAMON XE for DB2 PE identifiers with ORDER, the default order of PRIMAUTH and PLANNAME is used.

SERVER

Reported in the same column as REQUESTER. All server locations involved are listed in alphabetical sequence below the requester location.

TRACE NO

The number of the thread, in the format x.yyyyy, where x is the number of the TRACE subcommand and yyyyy is the number of the thread reported as a result of that TRACE subcommand.
STARTS PAGE NO
The number of the page on which the beginning of the thread is reported.
It is shown in the format x-yyyyy, where x is the location number and
yyyyy is the page number within the location.

Note: If more than one summary level is selected, STARTS PAGE NO
relates to the first summary printed for that thread.
Chapter 46. SQL Activity Report and Trace Blocks

Here you find the blocks reported by SQL Activity report and trace.

Note: In query CP and sysplex query parallelism, the TCB time in the reports and traces only reflects the TCB time of the originating record. For the TCB time of the parallel records, refer to the query parallelism workload detail block described in "Query Parallelism" on page 46-32.

"SQL Detail Section" on page 46-2
This section shows the report and trace detail portions for each SQL statement type.

"Report and Trace Details" on page 46-3
This section described the report and trace details for SQL activities.

"Workload Detail" on page 46-12
Workload detail is available on all summary levels. The workload figures are applied to the event being summarized.
SQL Detail Section

This section shows the report and trace detail portions for each SQL statement type.

Field descriptions are shown in “Field Descriptions of SQL Activity Detail Report and Trace Details -” on page 46-5.

DDF information is included in all SQL Activity summaries, when present.

When an SQL statement type is not recognized, other is printed.

The report and details blocks are divided into columns. The column labels vary for report and trace and for the summarization used. The following column labels are used:

NL  Nesting level of stored procedures, triggers and user-defined functions. This is valid for trace only. Values are summarized by occurrence.

EVENT  The event being reported or traced, such as the cursor name in a summary by cursor or the program name in a summary by program.

COUNT  The number of occurrences of this event.

TIMESTAMP  The timestamp of the event begin.

TOT.ELAPS  The total elapsed time of the event, that is the elapsed time for all statements within the event.

AET/EVENT  The average elapsed time of the event.

TOTAL TCB  The total TCB time of the event, that is the TCB time for all statements within the event.

TCB/EVENT  The average TCB time of the event.

ELAP.TIME  The elapsed time of the event.

TCB TIME  The TCB time of the event.
Report and Trace Details

This section described the report and trace details for SQL activities.

SQL Activity Trace Package Detail

The first line of PACKAGE details shows the fully-qualified package name, if applicable. It consists of:

- The location name of the DB2 subsystem where the package was bound
- The name of the package collection
- The name of the program
- The consistency token generated by the DB2 precompiler, if present
- The version ID of the package, if present

To avoid duplication, the package name is not embedded in the detail when the events are sorted in default sequence. It is printed once at the head of its work.

For details of other fields shown for package, see “Field Descriptions of SQL Activity Detail Report and Trace Details” on page 46-5.

Here is an example of an SQL Activity Trace Package Detail.

<table>
<thead>
<tr>
<th>NL</th>
<th>EVENT</th>
<th>TIMESTAMP</th>
<th>ELAP.TIME</th>
<th>TCB TIME</th>
<th>DETAIL</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

SQL Activity Trace Statement Detail

For field information, see “Field Descriptions of SQL Activity Detail Report and Trace Details” on page 46-5.

Here is an example of an SQL Activity Trace Statement Detail for CLOSE (DB2 10).

<table>
<thead>
<tr>
<th>NL</th>
<th>EVENT</th>
<th>TIMESTAMP</th>
<th>ELAP.TIME</th>
<th>TCB TIME</th>
<th>DETAIL</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Here is an example of an SQL Activity Trace Statement Detail (prior to DB2 10).

<table>
<thead>
<tr>
<th>NL</th>
<th>EVENT</th>
<th>TIMESTAMP</th>
<th>ELAP.TIME</th>
<th>TCB TIME</th>
<th>DETAIL</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
SQL Activity Trace Trigger Detail

For field information, see 'Field Descriptions of SQL Activity Detail Report and Trace Details -' on page 46-5.

Here is an example for an SQL Activity Trace Trigger Detail.

NL EVENT TIMESTAMP ELAP.TIME TCB TIME DETAIL
-------- ----------- ----------- --------- ---------------------------------------------------------------------
CLOSE 17:16:27.46 0.000458 0.000136 STMT# 66 CURSOR: C1 SQLSTATE: 00000 REOPTIMIZED(YES) KEEP UPDATE LOCKS : NO SCROLL(YES) SENSITIVE(YES) TABLE(STAT) IMPLICIT COMMIT(YES)

SQL Activity Trace Stored Procedure Detail and UDF Detail

For field information, see 'Field Descriptions of SQL Activity Detail Report and Trace Details -' on page 46-5.

Here is an example of an SQL Activity Trace Stored Procedure Detail.

NL EVENT TIMESTAMP ELAP.TIME TCB TIME DETAIL
-------- ----------- ----------- --------- ---------------------------------------------------------------------
CALL 17:16:27.46 0.180439 0.001409 STMT# 1216 PROCEDURE: MANFREDS STOPROC01 SQLSTATE: N/P SQLCODE: 0 COLLID : DSNTEP4 PROGRAM : DSNTEP4 SCHEMA : MANF_01 SCHEDULE TIME: N/P SCHEDULE TCB: N/P

Here is an example of an SQL Activity Trace UDF Detail.

NL EVENT TIMESTAMP ELAP.TIME TCB TIME DETAIL
-------- ----------- ----------- --------- ---------------------------------------------------------------------
INVOK 21:20:16.61 0.003687 FUNCTION : UDFSBSST2 SQLSTATE: N/P SQLCODE: 0 SCHEDULE TIME: N/P SCHEDULE TCB: N/P SCHEMA : USRT001 PACKAGE: STLEC1.USRT001.X'0000000000000001'

The following list shows the fields displayed in the SQL Activity detail report and trace details, in alphabetical order.
Field Descriptions of SQL Activity Detail Report and Trace Details -

The following list shows the fields displayed in the SQL Activity detail report and trace details, in alphabetical order.

**ACQUIRE**
The acquire level of the package showing USE or ALLOCATE.

**AET/OCCUR**
The average elapsed time for each occurrence.

**ACT_TIME**
Trigger activation time.

**BEFORE**

**AFTER**

**BUFFER WRITES**
The number of buffer writes.

**CLAIM COUNT**
The accumulated wait time for claim count.

**CLOSE TYPE**
The Close statement type. Possible values are:

- IMPLICIT
- EXPLICIT
- N/A

Otherwise the values are shown in hexadecimal.

**COLLID**
Collection identifier.

**COMMITS**
The total number of the following statements for the requester:

- Rollback
- Commit phase 2
- Sync

The total number of the following statements for the server:

- Commit request received
- Backout request received

**COUNT**
The number of occurrences as derived from the statement type.

**CURSOR**
The name of the cursor, if applicable.

**CURSOR_NAME**
Allocate cursor name.

**DBRM**
The name of the program, if applicable.

To avoid duplication, the DBRM name is not embedded in the detail when the events are sorted in default sequence. It is printed once at the head of its work.

**DRAIN LOCK**
The accumulated wait time for drain lock.
DYNAMICRULES
The value of the DYNAMICRULES option on the BIND/REBIND command:

RUN run time rules apply to a dynamic SQL statement for authorization checking and object qualification at run time.

BIND Bind-time rules apply to a dynamic SQL statement for authorization checking and object qualification at run time.

N/P DYNAMICRULES was not specified.

ELAP.TIME
Duration of trigger activity.

EVAL Triggered action condition. Possible values are:
• TRUE
• FALSE
• NONE

EXT_NAM
External trigger name.

FUNCTION
Function name.

GET PAGES
The number of Getpages.

GLOBAL LOCK
The accumulated wait time for global locks.

GRAN
Trigger granularity. Possible values are:
• ROW
• STMT

IMMEDWRITE
Indicates how DB2 updates group buffer pool dependent pages. This is only valid in a data sharing environment.

Group buffer pool dependent pages can be written to DASD or SYSTEM pagesets.

Values shown are:

NO DB2 uses normal write activity for updates, this is the default. Pages are written out at, or before phase 2 commit, or at the end of an abort for transactions that have rolled back.

PH1 Pages are written out at, or before phase 1 commit.

If a transaction subsequently rolls back, the pages are updated in the group buffer pool at the end of the rollback and are written out at the end of the abort.

YES Pages are written out to the coupling facility as soon as the buffer update commits. Pages are written out regardless of whether the update occurs during forward progress or rollback of the transaction.

This option can affect performance due to coupling facility overhead.
IMPLICIT COMMIT
The cursor attribute implicit commit, which closed the cursor. It can be YES or NO.

IN-DB2 CPU
The accumulated In-DB2 CPU time. This time includes CPU time that was consumed on an IBM specialty engine.

IN-DB2 ELAPSED
The accumulated In-DB2 elapsed time.

INDEX SCANS
The number of index scans.

ISOLATION or ISO
The isolation level of the statement:
- CS Cursor stability
- RS Read stability
- RR Repeatable read
- UR Uncommitted read

KEEPDYNAMIC
Indicates whether DB2 keeps (KEEPDYNAMIC(YES)) or discards (KEEPDYNAMIC(NO)) prepared SQL statements at commit points.

KEEP UPDATE LOCKS; KEEP UPD. LOCKS
Indicates whether X locks are used. X locks can only be used for SQL OPEN CURSOR statements and an isolation level of RS and RR.

LATCH
The accumulated wait for latch.

LOCAL
The number of statements that were distributed without going through VTAM.

LOCKS
The accumulated wait for locks.

# LOCATORS
The number of locators.

# LOCATOR_VAL
The number of locators.

LOG WRITER
The accumulated wait time for log writer.

NAME
The object name, without qualifier, in the DDL statement.

NEW DEGREE
The requested degree of parallelism regardless of whether the request is successful.

NEW SQLID
The requested SQL ID regardless of whether the request is successful.

OPTHINT
Value of optimization hints, if used.
PACKAGE
Package name (UDF only).

PAGE LATCH
The accumulated wait time for page latch.

PARALLEL GRP CREATES
The number of parallel groups creates.

PREPARE
Indicates whether the preparation of dynamic SQL statements was deferred:
  DEFER
  The preparation of the dynamic SQL statements that refer to remote objects was deferred until run time.
  NODEFER
  The dynamic SQL statements were prepared at bind time.

PREVIOUS DEGREE
The previous or current degree of parallelism.
  If the statement executed successfully, this is the previous degree of parallelism. If it executed unsuccessfully, this is the current degree of parallelism.

PREVIOUS SQLID
The previous or current SQL ID.
  If the request to change the SQL ID is successful, this is the user's previous SQL ID. If it is unsuccessful, this is the user's current SQL ID.

PROCEDURE
The unqualified stored procedure name.

PROC_LOC
Location of stored procedure.

PROC_NAME
Name of stored procedure.

PROC_QUALIF
Qualifier of stored procedure.

PROGRAM
Program or package name.

PROTOCOL
DB protocol:
  DRDA
  Convert three-part names to DRDA
  PRIVATE
  Three-part names use private protocol
  NOT_SPEC
  DB protocol was not specified.

READ-OTH. THREAD
The accumulated wait time for read activity by another thread.

RELEASE
The release level of the package, showing the option COMMIT or DEALLOCATE, if available.
REMOTE
The number of statements that went through VTAM.

REOPTIMIZED or REOPT
Indicates whether the access path of the SQL statement was reoptimized:
YES or Y
The access path was reoptimized at run time.
NO or N
The access path was only optimized at bind time.

RID-LIMIT EXC.
The number of times RID list was not used because the number of RIDs would have exceeded DB2 limits.

RID-NO STORAGE
The number of time a RID list was not used because there is not enough storage available to hold the list of RIDs.

ROWS EXAMINED
The number of rows examined.

ROWS PROCESSED
The number of rows processed.

SCHEDULE TCB
The TCB time for scheduling the stored procedure.

SCHEDULE TIME
The elapsed time for scheduling the stored procedure. This field also includes the time for processing application logic, if any, up to the first SQL statement within the stored procedure.

SCHEMA
Schema name.

SCROLL
Identifies the cursor scrollability. It can be one of the following:
• Scroll
• None-scroll

SENSITIVE
The cursor sensitivity. It can be one of the following:
• Sensitive
• Insensitive
• Unspecified

SERVER
The server location in a distributed transaction. If there are multiple server locations, an asterisk (*) is printed.

SORTS
The number of sorts.

SQLCODE or SQLCO
The return code from the SQL event. Obtained from the DB2 SQLCODE which remaps the DB2 field.

SQLSTATE or SQLST
The SQL state.
STATUS
The status of the statement:
• SUCCESSFUL
• FAILED
• SYSADM — Although authorization validation failed, the statement is successful because the user had SYSADM authority.

STMT
Triggering SQL statement.
• UPDATE
• INSERT
• DELETE

STMT#
The number of the statement executed.

Note: For implicit connects, the statement number shown is the number of the SQL statement that caused the connect.

STMT ID
The statement identifier.

STMT TYPE
The statement type.

Note: In a summary by statement number, you find the statement number instead of the statement type.

SYNC BUFF READS
The number of synchronous buffer reads.

SYNCHRON. I/O
The accumulated wait for synchronous I/O.

TABLE
The cursor result table type. It can be one of the following:
• Static
• Dynamic
• Unspecified

TABLESPACE SCANS
The number of tablespace scans.

TCB/OCCUR
The average TCB time for each occurrence. It is the TCB time spent at the location being traced or reported on this line. For requester locations, it shows only the small amount of processing done at the requester.

TCB TIME
Duration of trigger activity.

TEXT
The text of the SQL statement, if present.

Note:
1. The text is only printed in a summary by occurrence if it is a dynamic SQL.
2. Text exceeding 5000 characters is truncated.
3. Host variables are presented as :H.
**TIMESTAMP**
Trigger timestamp.

**TRIGGER**
Trigger name.

**TYPE**  The type of object in the DDL statement.

**UNIT SWITCH**
The accumulated wait time for synchronous execution unit switches.

**WRITE-OTH. THREAD**
The accumulated wait time for write activity by another thread.
Workload detail is available on all summary levels. The workload figures are applied to the event being summarized.

Any workload performed during thread creation is shown on the first SQL statement occurrence encountered in a thread.

“Accounting” on page 46-13
The layout of the accounting section is identical to the accounting long report or trace, depending on whether an SQL Activity report or trace has been requested.

“Data Capture Activity” on page 46-14
This block shows the average data capture activity performed by the event.

“Exit Activity” on page 46-15
This block shows the exits performed by the event.

“Function Resolution Activity” on page 46-16
This section shows the layout of a function resolutions block and its field descriptions.

“I/O Activity” on page 46-18
This block shows the I/O activity for each object performed by the event.

“Lock Suspension Activity” on page 46-20
This block shows the lock suspension activity for each object performed by the event.

“Minibind Activity” on page 46-22
The minibind activity block shows information about mini plans, which are generated by the optimizer at bind and SQL prepare time. This block is written once for each IFCID 022 encountered. The block consists of a header followed by one or more repeating sections.

“Page and Row Locking Activity” on page 46-29
This block shows the page locking, row locking, and lock avoidance activity for each object, performed by the event.

“Query Parallelism” on page 46-32
This section introduces the Query Parallelism block.

“RID List Processing” on page 46-34
This block shows the record ID (RID) list activity performed by the event.

“Scan Activity” on page 46-35
This block shows the total scan activity for each object, performed by the event.

“Sort Activity” on page 46-37
This block shows sort activity for each sort performed by the event.

“Host Variables” on page 46-43
This block shows the host variables to represent the values that will be sent to or received from the SQL statement.

“Workload Highlight” on page 46-46
This block shows the highlights of the workload activity performed by the event. All workload fields available in the SORTBY option are included.
Accounting

The layout of the accounting section is identical to the accounting long report or trace, depending on whether an SQL Activity report or trace has been requested.

For explanations of the blocks and fields shown in the accounting section, see “Accounting Report and Trace Blocks” on page 5-43.
**Data Capture Activity**

This block shows the average data capture activity performed by the event.

**Data Capture Workload Block Example**

Here is an example of the data capture workload block.

```
--- DATA CAPTURE ----------------------------------------------------------------

DESCRIPTION : 3.14 MAX READ TIME: 1.928397 DATA DESC RETURNED: 3.77
AET/DESCRIPTION : 0.028367 RECS RETURNED: 24.86 TABLES RETURNED : 0.00
LOG READS : 5.20 RECS CAPTURED: 29.15
AET/EXTRACTION: 1.044382 ROWS RETURNED: 132.50
```

**Field description**

Here is a description of the field labels shown in the data capture workload block:

**DESCRIPTION**

The average number of data capture describes.

**AET/DESCRIPTION**

The average elapsed time of data capture describes.

**LOG READS**

The average number of log reads performed.

**AET/EXTRACTION**

The average elapsed time of log extraction.

**MAX READ TIME**

The longest elapsed time of a log read.

**RECS RETURNED**

The average number of log records returned.

**RECS CAPTURED**

The average number of records that were captured for this update. To perform all data capture updates, all captured log records need to be returned.

**ROWS RETURNED**

The average number of data rows returned.

**DATA DESC RETURNED**

The average number of data capture data descriptions returned.

**TABLES RETURNED**

The average number of data capture tables returned.
Exit Activity

This block shows the exits performed by the event.

Exits Workload Block Example

Here is an example of an exits workload block:

<table>
<thead>
<tr>
<th>MEMBER</th>
<th>VALIDATION TOTAL</th>
<th>AET/EXIT</th>
<th>EDIT TOTAL</th>
<th>AET/EXIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>SE11</td>
<td>1</td>
<td>N/C</td>
<td>0</td>
<td>0.000060</td>
</tr>
</tbody>
</table>

Field description

Here is a description of the field labels shown in the exits workload block:

MEMBER
The name of the DB2 member within the DB2 data sharing group.

VALIDATION TOTAL
The number of results of a validation exit call written for every validation row.

VALIDATION AET/EXIT
The summarized elapsed validation time divided by the value in VALIDATION TOTAL.

EDIT TOTAL
The summary of results of an edit exit call to encode a record written for every row edited and the results of an edit exit call to decode a record written for every row decoded.

EDIT AET/EXIT
The summarized elapsed edit time divided by the value in EDIT TOTAL.
SQL Activity - Function Resolution

Function Resolution Activity

This section shows the layout of a function resolutions block and its field descriptions.

SQL Activity Function Resolutions Workload Block

The following figure shows the layout of the function resolutions block:

--- FUNCTION RESOLUTION(S) -----------------------------------------------------------------------------------------------
| QUERYNO : 1383 | PLANNAME : DSNTEP61 | COLLECTION_ID : DSNTEP61 |
| APPLNAME : xxxxxxxx | PROGNAME : xxxxxxxx | CONSIS_TOKEN : xxxxxxxxxxxxxxxx |
| BIND_TIME: 01/30/10 03:28:55.21 | VERSION : xxxxxxxxxxxxxxxxxxxxx |
| CURRENT_PATH : ........10........20........30........40........50........60........70........80........90.......100 |
| ........110.......120.......130.......140.......150.......160.......170.......180.......190.......200 |
| ........210........220........230........240........254 |
| FUNCT_SCHEMA : xxxxxxxx | FUNCT_NAME : xxxxxxxxxxxxxxxxxx | SPECIFIC_NAME : xxxxxxxxxxxxxxxxxx |
| FUNCT_TYPE : xxxxx |
| VIEW_CREATOR : NAME-111 | VIEW_NAME : xxxxxxxxxxxxxxxxxx |
| QUERY_BLOCKNO : 53 |
| FUNCT_TEXT : ........10........20........30........40........50........60........70........80........90.......100 |
| ........110.......120.......130.......140.......150.......160.......170.......180.......190.......200 |
| ........210........220........230........240........254 |

Field description

Here is a description of the field labels shown in the function resolutions block:

QUERYNO
The query number.

PLANNAME
The plan name.

COLLECTION_ID
The collection ID.

APPLNAME
The name of the application.

PROGNAME
The program name.

CONSIS_TOKEN
The consistency token.

BIND_TIME
The time stamp of the bind time.

VERSION
The version ID.

CURRENT_PATH
The current path.

FUNCT_SCHEMA
A short SQL identifier, either ordinary or delimited, following the concept of qualified names consistent with the ANSI/ISO SQL92 standard.

FUNCT_NAME
The name of a function without a qualifier.
SQL Activity - Function Resolution

SPECIFIC_NAME
Identifies the particular function. The specific name must identify a specific function name in the explicitly or implicitly specified schema.

FUNCT_TYPE
The classification of the function:

SCALAR  Scalar UDF
TABLE    Table UDF

VIEW_CREATOR
The name of the view creator if the function is referenced in a view definition.

VIEW_NAME
The name of the view if the function is referenced in a view definition.

QUERY_BLOCKNO
A number that identifies the query block number being explained.

FUNCT_TEXT
Contains the text of the function reference, function name, and parameters.
It can be up to 254 characters long.
SQL Activity - I/O

I/O Activity

This block shows the I/O activity for each object performed by the event.

I/O Activity Workload Block Example

The following example shows the I/O activity workload block.

--- I/O ACTIVITY ---------------------------------------------------------------------------------------------------------------------------------------------------
<table>
<thead>
<tr>
<th>DATABASE</th>
<th>PAGESET</th>
<th>MEMBER</th>
<th>BP</th>
<th>TOTAL</th>
<th>AET</th>
<th>TYPE</th>
<th>AET/WITH</th>
<th>%WITH</th>
<th>PAGE/WITH</th>
<th>%WITHOUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBPARALL</td>
<td>TSPPARALL</td>
<td>SE12</td>
<td>BP4</td>
<td>3</td>
<td>0.1296</td>
<td>3</td>
<td>SYNCH</td>
<td>0.129597</td>
<td>100.00</td>
<td>1.00</td>
</tr>
<tr>
<td>WRKSE12</td>
<td>DSNK01</td>
<td>SE12</td>
<td>BP0</td>
<td>102</td>
<td>0.0164</td>
<td>102</td>
<td>SYNCH</td>
<td>0.016358</td>
<td>100.00</td>
<td>1.00</td>
</tr>
</tbody>
</table>

Field description

Here is a description of the field labels shown in the I/O activity workload block:

DATABASE
The database name. If the name is not available, the decimal DBID/OBID is printed.

PAGESET
The page set name. If the name is not available, the decimal DBID/OBID is printed.

MEMBER
The name of the DB2 member within the DB2 data sharing group. This field shows N/P in a non-data-sharing environment.

BP
The buffer pool name.

I/O REQUEST TOTAL
The total number of I/O requests.

I/O REQUEST AET
The average elapsed time for each I/O request.

READ REQUEST TOTAL
The number of read I/O requests of a specific type.

READ REQUEST TYPE
The type of read request:

SYNCH
Synchronous read request

SEQPF
Sequential prefetch request

DYNPF
Dynamic prefetch request

LSTPF
List prefetch request

READ REQUEST AET/WITH
The average elapsed time for a read with I/O of a specific type.

READ REQUEST %WITH
The percentage of total read requests with I/O for a particular type.

READ REQUEST PAGE/WITH
The pages read for each read request with I/O of a particular type.
SQL Activity - I/O

**READ REQUEST %WITHOUT**
The percentage of total read requests without I/O for a particular type. This can occur because all the pages requested by a prefetch read were already in the buffer pool.

**WRITE REQUEST TOTAL**
The number of write I/O requests.

**WRITE REQUEST TYPE**
The type of write request.

**WRITE REQUEST CAST**
Indicates whether the write operations were initiated due to a coupling facility castout.

**WRITE REQUEST AET**
The average elapsed time for each write.

**WRITE REQUEST PAGE/WRITE**
The number of pages written.
# SQL Activity - Lock Suspension

## Lock Suspension Activity

This block shows the lock suspension activity for each object performed by the event.

## Lock Suspension Activity Workload Block Example

The field labels shown in the following sample layout of "Lock Suspension Activity Workload Block" are described in the following section.

--- LOCK SUSPENSION ACTIVITY -------------------------------------------------------------------------------------------------
---------- SUSPEND REASON ---------- NORML RESUME TIMEO RESUME DEADL RESUME
RESOURCE NAME TYPE REQUEST LOCAL LATCH IRLMQ GROUP NOTIF OTHER COUNT AET COUNT AET COUNT AET
MEMBER
DBPARALL TSPARALL DATAPAGE NOTIFY 0 0 0 24 24 0 24 0.74382 0 N/C 0 N/C
SE11
DBPARALL TSPARALL DATAPAGE LOCK 0 3 0 0 0 0 3 0.04096 0 N/C 0 N/C
SE11
DBPARALL TSPARALL DATAPAGE LOCK 0 5 0 0 0 0 5 0.06957 0 N/C 0 N/C
SE12
DBPARALL TSPARALL DATAPAGE UNLOCK 0 1 0 2 2 0 3 0.59058 0 N/C 0 N/C
SE21
---

The following list describes the fields in the lock suspension activity workload block:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| RESOURCE NAME | The name of the resource on which the suspended request is made. The content of the field depends on the resource type:  
  - The plan name for SKCT  
  - The collection and package IDs for SKPT  
  - The collection ID for COLLECT  
  - The database name for DATABASE, CDB PLK, DBD PLCK  
  - The buffer pool ID for ALTERBUF, GBP S/S, P/P PLCK, PAGEPLCK, GBP CAST, P/P CAST  
  - The anchor point ID for HASH-ANC  
  - The row ID for ROW  
  - N/A for MASS, UTILITY, BINDLOCK, ALTERBUF, CATM MIG, CATM CAT, CATM DIR  
  - The database and page set names for all others  
The database and page set names are translations obtained from the IFCIDs 105 and 107. If these records are unavailable, the decimal DBIDs and OBIDs are printed. |
| MEMBER      | The name of the DB2 member within the DB2 data sharing group. |
| TYPE        | The type of the locked resource. Possible values are shown in Table 33-1 on page 33-3 |
| REQUEST     | The type of request that has been suspended:  
  - LOCK  IRLM lock request  
  - UNLOCK  IRLM unlock request  
  - CHANGE  IRLM change request |
SQL Activity - Lock Suspension

**QUERY**
IRLM query request

**NOTIFY**
IRLM notify request

**DRAIN**
Drain request

**LATCH**
Latch request

**SUSPEND REASON LOCAL**
The number of suspensions due to local resource contentions.

**SUSPEND REASON LATCH**
The number of suspensions due to IRLM latch contentions.

**SUSPEND REASON IRLMQ**
The number of suspensions due to IRLM queued requests.

**SUSPEND REASON GROUP**
The number of suspensions due to global contention.

**SUSPEND REASON NOTIFY**
The number of suspensions due to intersystem message sending.

**SUSPEND REASON OTHER**
The number of suspensions due to reasons other than those listed previously.

*Note:* For drain suspensions, the suspension reason is always “waiting for the claim count to reach zero” and is categorized as OTHER.

**NORML RESUME COUNT**
The number of suspensions that ended in the task, resuming normal processing after the lock request has completed.

**NORML RESUME AET**
The normal resume average elapsed time. This is the normal resume elapsed time divided by the NORML RESUME COUNT.

**TIMEO RESUME COUNT**
The number of suspensions that ended in a timeout.

**TIMEO RESUME AET**
The average elapsed timeout time. This is the elapsed timeout time divided by the TIMEO RESUME COUNT.

**DEADL RESUME COUNT**
The number of suspensions that ended in a deadlock.

*Note:* Drain suspensions do not end in a deadlock.

**DEADL RESUME AET**
The average elapsed deadlock time. This is the elapsed deadlock time divided by the DEADL RESUME COUNT.
**Minibind Activity**

The minibind activity block shows information about mini plans, which are generated by the optimizer at bind and SQL prepare time. This block is written once for each IFCID 022 encountered. The block consists of a header followed by one or more repeating sections.

One mini plan is generated for each table and for each subselect block in the query. This means that if your query uses subqueries, more than one mini plan record is written.

**Note:**
1. This block is shown for SQL Activity trace only.
2. When interpreting this record, relate table and mini plans by table name.
3. The order of the mini plans might not be the same as the order of the table as written in the SQL statement.
4. When you are not sure about the accessing order of the tables, use EXPLAIN to get the query block number and plan number.
5. This block also shows whether sequential prefetch is used.

If the IFCIDs 105 and 107 are present before IFCID 022, the DBID and OBID can be translated.

**Explanation of long names**

DB2 supports long values, up to 128 characters, for certain fields. When this happens, the value shown by OMEGAMON XE for DB2 PE in the report block can be truncated, depending on the space available. Truncated values are then listed at the end of each logical report unit, together with the full values.

When a value is truncated, the "greater than" sign (>) is printed instead of a colon (:) following the label name. The full value starts with a "greater than" sign followed by the label. For example:

```
Tname  > This value
...    
...    
>Tname  : This value was truncated
...    
```

If truncated values are listed, the "greater than" sign (>) is shown at the end of each value, because there is no colon (:) as a delimiter between the label and the value. In lists the label is used as a column heading.

For example:

```
<table>
<thead>
<tr>
<th>ACCESS_CREATOR</th>
<th>ACCESS_NAME</th>
<th>MATCHCOLS</th>
<th>INDEXONLY</th>
<th>PREFETCH_INDEX</th>
<th>OPERATION</th>
<th>MIXOPSEQ</th>
</tr>
</thead>
<tbody>
<tr>
<td>TDK_LONG&gt;</td>
<td>IX_OMPE_FIRST_LONG&gt;</td>
<td>0</td>
<td>YES</td>
<td></td>
<td>SEQUENTIAL</td>
<td>SCAN 1</td>
</tr>
<tr>
<td></td>
<td>ix_ompe_first_long&gt;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>IX_OMPE_FIRST_LONG&gt;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>IX_OMPE_FIRST_LONG&gt;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

The mapping between truncated and full values remains the same for multiple reports from the same input data. This mapping for multiple reports from different input data cannot be guaranteed.

**Minibind Workload Block Example**

Here is an example of the minibind workload block.
Field description

Here is a description of the field labels shown in the minibind workload block.

**QUERYNO**
The number identifying the statement to be prepared.

**PLANNAME**
The plan name or package ID.

**COST**
The relative cost of the SQL statement. It might not relate to the actual CPU or elapsed time for the query.

**PARALLELISM_DISABLED**
Indicates whether query parallelism is disabled by the resource limit facility (RLF) for dynamic queries:

- **NO** The RLF does not affect this statement.
- **I/O ONLY** Query I/O parallelism is disabled.
- **CP ONLY** Query CP parallelism is disabled.
- **CP + I/O** Query I/O and CP parallelism is disabled.
- **X** Sysplex query parallelism is disabled.
**SQL Activity - Minibind**

- **X + I/O**
  - Sysplex query and query I/O parallelism is disabled.

- **X + CP**
  - Sysplex query and query CP parallelism is disabled.

- **YES**
  - The entire query parallelism (I/O, CP, and Sysplex) is disabled.

- **N/A**
  - Query parallelism does not apply to this statement.

**QBLOCKNO**
- The position of the query in the statement.

**COLLID**
- The collection ID of the package.

**PROGNAME**
- The name of the package containing the statement to be prepared.

**CONSISTENCY_TOKEN**
- The consistency token.

**APPLNAME**
- The name of the application plan.

**WHEN_OPTIMIZE**
- Indicates when the access path of the SQL statement is optimized and determined:
  - **BIND**
    - The access path is determined at bind and run time.

- **DEFAULT**
  - The access path is determined at bind time.

- **REOPT**
  - The statement is bound with REOPT. The access path is determined at run time.

- **RUN**
  - The access path is determined at run time.

**OPT_HINT_IDENT**
- Access path hint value.

**OPTIMIZE_HINTS_USED**
- Indicates whether the query used access path hints.

**UNITS**
- Cost in CPU units.

**MILLI_SEC**
- Cost in milliseconds.

**COST_CATEGORY**
- Cost category.

**BIND_TIME**
- The date and time at which the plan or package to which the SQL statement belongs was bound.

**VERSION**
- The version ID of the package (64 characters).

**PLANNO**
- The number of the step in which the query is processed.

**METHOD**
- The join method used for the step.
**SORTN_UNIQ**
Indicates whether the new table is sorted to remove duplicate rows.

**SORTC_UNIQ**
Indicates whether the composite table is sorted to remove duplicate rows.

**DATABASE**
The database ID.

**NEXTSTEP**
The next step in a join.

*NOT APPLICABLE* is printed if this is the last step of a join, or if this is not a join.

**SORTN_JOIN**
Indicates whether the new table is sorted for a merge scan join or hybrid join. For a hybrid join, this is a sort of the RID list.

**SORTC_JOIN**
Indicates whether the composite table is sorted for a nested loop join, merge scan join, or hybrid join.

**OBJECT**
The internal ID of the table space.

**ACCESTYPE**
The method of accessing the new table. *N/P* is printed if there is no access type.

**SORTN_ORDERBY**
Indicates whether the new table is sorted for ORDER BY.

**SORTC_ORDERBY**
Indicates whether the composite table is sorted for ORDER BY.

**CREATOR**
The creator of the new table accessed in this step.

**PAGE_RANGE**
Indicates whether the table qualifies for page range screening so that plans scan only the partitions that are needed.

**SORTN_GROUPBY**
Indicates whether the new table is sorted for GROUP BY.

**SORTC_GROUPBY**
Indicates whether the composite table is sorted for GROUP BY.

**TNAME**
The name of the table accessed in this step, without qualifier. This field is blank if a view is used instead of a real table.

**JOIN_TYPE**
The type of join enabled:
- **LEFT** Left outer join
- **FULL** Full outer join
- **INNER** Inner join
- **STAR** Star join
SQL Activity - Minibind

N/A   Not applicable is shown if DB2 never produces a counter value in a specific context.

SORTN_PGROUP_ID
The parallel group ID for the parallel sort of the new table.
A parallel group is the collective term for consecutive operations (in this case a sort) executed in parallel that have the same number of parallel tasks.

SORTC_PGROUP_ID
The parallel group ID for the parallel sort of the composite table.

CORRELATION_NAME
The correlation name of a table or view that is specified in the statement. If no correlation name is specified, the field is blank.

MERGE_JOIN_COLS
The number of columns that are joined during a merge scan join.

ACCESS_DEGREE
The number of parallel tasks or operations activated by a query.

JOIN_DEGREE
The number of parallel tasks or operations used in joining the composite table with the new table.

TSLOCKMODE
Indicates the lock mode to be acquired on the new table or its table space.
If the isolation can be determined at bind time, possible values are:

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IS</td>
<td>Intent share lock</td>
</tr>
<tr>
<td>IX</td>
<td>Intent exclusive lock</td>
</tr>
<tr>
<td>S</td>
<td>Share lock</td>
</tr>
<tr>
<td>U</td>
<td>Update lock</td>
</tr>
<tr>
<td>X</td>
<td>Exclusive lock</td>
</tr>
<tr>
<td>SIX</td>
<td>Share with intent exclusive lock</td>
</tr>
<tr>
<td>N</td>
<td>UR isolation, no lock</td>
</tr>
</tbody>
</table>

If the isolation cannot be determined at bind time, the lock mode determined by the isolation at run time is shown by the following values:

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NS</td>
<td>For UR isolation: no lock. For CS or RR isolation: an S lock.</td>
</tr>
<tr>
<td>NIS</td>
<td>For UR isolation: no lock. For CS or RR isolation: an IS lock.</td>
</tr>
<tr>
<td>NSS</td>
<td>For UR isolation: no lock. For CS isolation: an IS lock. For RR isolation: an S lock.</td>
</tr>
<tr>
<td>SS</td>
<td>For UR or CS isolation: no lock. For RR isolation: an S lock.</td>
</tr>
</tbody>
</table>

PARALLELISM_MODE
The kind of parallelism used at bind time:

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>I/O</td>
<td>Query I/O parallelism</td>
</tr>
<tr>
<td>CP</td>
<td>Query CP parallelism</td>
</tr>
<tr>
<td>X</td>
<td>Sysplex query parallelism</td>
</tr>
<tr>
<td>NO</td>
<td>No parallelism was used.</td>
</tr>
</tbody>
</table>
ACCESS_PGROUP_ID
The ID of the parallel group for accessing the new table.

JOIN_PGROUP_ID
The ID of the parallel group for joining the composite table with the new table.

ACCESS_NAME
The index name. This field applies only to index scans. N/A is printed for table space scans or when no index is used.

ACCESS_CREATOR
The index creator.

STATEMENT_CACHE
Statement cache flag. Possible values are:
- YES The prepared statement is retrieved from the prepared statement cache.
- NO The prepared statement is not retrieved from the prepared statement cache.

MATCHCOLS
The number of index keys used in an index scan. This field is 0 if either no index is used or an index is used that has no matching columns.

PREFETCH
Indicates what kind of prefetch of the data is used:
- SEQ Sequential prefetch
- LIST List prefetch
- No No prefetch

OPERATION
The type of index access operation.

PREFETCH_INDEX
Indicates whether data pages are to be read in advance by a prefetch.

MIXOPSEQ
The sequence number of a step in a multiple index operation.

INDEXONLY
Indicates whether the access to an index alone is sufficient to carry out the step.

COLUMN_FN_EVAL
Indicates when an SQL column function is evaluated.

PAGES_FOR_TABLE
Pages for table.

TAB_CARDINALITY
Table cardinality.

DIRECT_ROW_ACC
Indicates whether direct row access was used, possible values are:
- YES Direct row access was used
- NO Direct row access was not used

STARJOIN
Indicates whether star join was used, possible values are:
### SQL Activity - Minibind

<table>
<thead>
<tr>
<th>YES</th>
<th>Star join was used</th>
</tr>
</thead>
<tbody>
<tr>
<td>NO</td>
<td>Star join was not used</td>
</tr>
<tr>
<td>N/A</td>
<td>Not applicable</td>
</tr>
</tbody>
</table>
Page and Row Locking Activity

This block shows the page locking, row locking, and lock avoidance activity for each object, performed by the event.

The page and row locking activity block is only printed if a commit occurred or a thread terminated.

In summary by occurrence, page and row locking activity information generated for explicit commits is shown on the relevant commit events.

In summaries by cursor or program, any explicit commits occurring during the life of that cursor or program are counted. Page and row locking activity caused by those commits is shown on the relevant cursor or program.

In summaries by statement number or statement type, commits are not counted. Because page and row locking activity is not relevant for these summary levels, it is not printed.

Any page or row locking activity occurring when a thread terminated is shown in the summary by thread. This activity is added to any page or row locking which took place in the body of the thread. Therefore, page and row locking figures in summary by thread can be greater than the sum of page locking figures shown in the body of the thread. The difference is the page and row locking activity occurring at thread termination.

**Page and Row Locking Workload Block Example**

Here is an example of the page and row locking workload block.

<table>
<thead>
<tr>
<th>MEMBER</th>
<th>DATABASE</th>
<th>PAGESET</th>
<th>PAGE</th>
<th>SIZE</th>
<th>MAXIMUM PAGE</th>
<th># LOCK OR ROW LOCKS</th>
<th>HIGHEST LOCK</th>
<th>TS</th>
<th>LOCK ESCAL</th>
<th>LOCK AVOID</th>
</tr>
</thead>
<tbody>
<tr>
<td>SE11</td>
<td>DBPARALL</td>
<td>TSPARALL</td>
<td>1</td>
<td></td>
<td>1</td>
<td>0</td>
<td>SPL</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SUMMARY: MAX PAGE OR ROW LOCKS HELD</td>
<td>1</td>
<td>LOCK ESCALATIONS</td>
<td>: SHARED</td>
<td>0</td>
<td>EXCLUSIVE</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SE12</td>
<td>DBPARALL</td>
<td>TSPARALL</td>
<td>2</td>
<td></td>
<td>5</td>
<td>0</td>
<td>SPL</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SUMMARY: MAX PAGE OR ROW LOCKS HELD</td>
<td>5</td>
<td>LOCK ESCALATIONS</td>
<td>: SHARED</td>
<td>0</td>
<td>EXCLUSIVE</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SE21</td>
<td>DBPARALL</td>
<td>TSPARALL</td>
<td>1</td>
<td></td>
<td>2</td>
<td>0</td>
<td>SPL</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SUMMARY: MAX PAGE OR ROW LOCKS HELD</td>
<td>2</td>
<td>LOCK ESCALATIONS</td>
<td>: SHARED</td>
<td>0</td>
<td>EXCLUSIVE</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TOTAL</td>
<td></td>
<td></td>
<td>4</td>
<td></td>
<td>4</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Note:**

1. The DBID and OBID are obtained from IFCID 020.
2. The values in MAX PAGE OR ROW LOCKS HELD, LOCK ESCALATIONS SHARED, and LOCK ESCALATIONS EXCLUSIVE are accumulated within a subsystem. They are reset only at thread deallocation or when a new user signon occurs.
3. The values in MAXIMUM PAGE OR ROW LOCKS, HIGHEST LOCK, and # LOCK ESCAL are reset at commit time for dynamic BINDs and for static BINDs for which release (commit) is specified. Otherwise, these values accumulate until thread deallocation or until a new user signon occurs.
4. IFCID 218 is an additional lock summary record, written for lock avoidance. It indicates whether a successful lock avoidance test occurred during a given unit of work. The record is externalized for the agent at each commit or rollback.
5. For each event, the relevant IFCID 020 and 218 records are processed. If there is a DBID/OBID combination present for IFCID 218 but not for IFCID 020, the
SQL Activity - Page and Row Locking

IFCID 020 fields show N/P. If there is a DBID/OBID combination present for IFCID 020 but not for IFCID 218, the IFCID 218 field (LOCK AVOID SUCCESSFUL) shows N/P.

Field description

Here is a description of the field labels shown in the page and row locking workload block.

MEMBER
The name of the DB2 member within the DB2 data sharing group.

DATABASE
The database name, if available.
If the name is not available, the decimal DBID is printed instead.

PAGESET
The page set name, if available.
If the name is not available, the decimal OBID is printed instead.

COUNT
The number of page locking or row locking occurrences for each page set.
  • Specific database and page set:
    - At commit time: always 1
    - At thread termination: the number of times this database and page set occurred on a commit record
  • TOTAL
    - At commit time: the total number of page sets listed
    - At thread termination: the sum of the values for all page sets

LOCK SIZE
The lock size used:
PAGE  Page lock
ROW   Row lock
TABLE  Table space or table lock
LOB   LOB lock
UNKN
  • Unknown lock
  * Multiple lock sizes

MAXIMUM PAGE OR ROW LOCKS
The maximum number of either page locks or row locks held at one time against this object.

# LOCK ESCAL
The number of lock escalations:
  • 0 if no escalations occur
  • For simple table spaces and partitioned table spaces not using selective partition locking (SPL): 1 if any escalation occurred for this table space in this logical unit of work
  • For segmented table spaces: the number of tables within the table space that have experienced lock escalations
  • For partitioned table spaces using SPL: the number of partitions for which locks escalated within the table space
SQL Activity - Page and Row Locking

The TOTAL contains the sum of all values in this column.

HIGHEST LOCK
The highest table space lock state.
If the table space is simple or partitioned not using SPL, it is the highest lock state for this database or page set. At trace end, it is the largest value from any commit for this object. The following values are possible:

- **IS** Intent share
- **IX** Intent exclusive
- **S** Share
- **U** Update share
- **SIX** Share with intent exclusive
- **X** Exclusive
If the table space is segmented or partitioned using SPL, this field is blank.

TS TYPE
The table space type:

- **SIMPL** Simple table space
- **SEG** Segmented table space
- **PARTI** Partitioned table space
- **SPL** Partitioned table space using selective partition locking (SPL)
- **LOB** LOB table space

LOCK AVOID SUCCESSFUL
Indicates whether there was a successful lock avoidance test during the unit of work.
If the state of this field changed during the summarization period, an asterisk (*) is shown.

MAX PAGE OR ROW LOCKS HELD
The maximum number of page locks and row locks held at one time across all objects.

LOCK ESCALATIONS: SHARED
The total of shared lock escalations.

LOCK ESCALATIONS: EXCLUSIVE
The total of exclusive lock escalations.
SQL Activity - Query Parallelism

Query Parallelism
This section introduces the Query Parallelism block.

This block shows query parallelism activity performed by the event.

Note: In query CP and sysplex query parallelism, this is the only place where the TCB time of the parallel records is shown.

Query Parallelism Workload Block Example
Here is an example of a Query Parallelism Workload block.

```
--- QUERY PARALLELISM -----------------------------------------------------------
QUERY BLOCK PARALLEL GROUP AT BIND AT RUN AT RUN REASON ELAPSED TIME CPU TIME TYPE MEMBERS
1 1 3 3 3 NORMAL 0.895716 0.043467 CP 3
```

Field description
An example of a query parallelism workload block is shown in "Query Parallelism Workload Block Example."
The fields in the query parallelism workload block are:

**QUERY BLOCK**
The query block number.

**PARALLEL GROUP**
The parallel group number.

**PLANNED AT BIND**
The degree of parallelism planned at bind time.

This field contains 0 if host variables in the statement caused the parallelism decision to be made at bind time.

**PLANNED AT RUN**
The degree of parallelism planned at run time.

**NEGOTIATED AT RUN**
The degree of parallelism negotiated at run time, which depends on buffer pool availability.

If the value in this field is 1, the plan for parallel I/O processing falls back to sequential execution mode.

**REASON**
The reason for deriving the planned run time degree of parallelism:

**NORMAL**
The planned run time degree is derived from planned bind-time degree.

**HOSTVAR**
Host variable partitioning

**NOESA**
No ESA sort support

**CURSOR**
Cursor that can be used for update and delete.

**EMPTY**
Empty parallel group
SQL Activity - Query Parallelism

ENCLUNAV
MVS/ESA enclave services are not available

UNKNOWN
None of the above

PIPE ELAPSED TIME
The time of pipe creation subtracted from the time of pipe termination.

TASK CPU TIME
The sum of the normalized CPU times spent for the parallel tasks. In sysplex query parallelism, the CPU times are normalized by the conversion factor that is derived from IFCID 106 and related to the conversion factor of the originating task.

If IFCID 106 is not present, asterisks are printed.

The task CPU time is calculated as follows:

- Let $C_{V_o}$ be the conversion factor for the member where the originating thread is running.
- Let $C_{V_p}$ be the conversion factor for the member where the parallel thread is running.
- Let $TCB_p$ be the TCB time that is recorded by DB2 for an activity of the parallel thread.
- Then the following formula applies:
  
  Normalized TCB time for that activity = $(TCB_p \times (C_{V_o} / C_{V_p}))$

TYPE
The type of parallelism:

- CP CP parallelism
- I/O I/O parallelism
- SYS Sysplex query parallelism

NUMBER OF MEMBERS
The number of members on which the query executed.
### SQL Activity - RID List Processing

**RID List Processing**

This block shows the record ID (RID) list activity performed by the event.

**RID List Processing Workload Block Example**

Here is an example of the RID list processing workload block.

--- RID LIST PROCESSING ----------------------------------------------------------------------------------------
RIDS IN FINAL LIST: 38 RID LIST USED: 2 UNUSED (LIMIT EXCEEDED): 5 UNUSED (NO STORAGE): 1
DATABASE PAGESET THRESHOLD RIDS OBTAINED RIDS EXCEEDED LIMIT
NHDBASE1 NHINDEX1 4075 36 3
NHDBASE2 NHINDEX2 36000 87 2
AVERAGE 20037.50 61.50 2.50

**Field description**

Here is a description of the field labels shown in the RID List Processing Workload Block.

- **RIDS IN FINAL LIST**
  - The number of RIDs in the final list.

- **RID LIST USED**
  - The number of times RID list was used.

- **UNUSED (LIMIT EXCEEDED)**
  - The number of RID lists not used because the number of RIDs exceeded the maximum limit.

- **UNUSED (NO STORAGE)**
  - The number of RID lists not used because no RID storage was available.

- **DATABASE**
  - The database name for the index.

- **PAGESET**
  - The internal identifier index fan-set descriptor for the index.

- **THRESHOLD**
  - The threshold value for the index.
  - The threshold value for list prefetch and ORing multiple indexes for access is the maximum of 25 percent of the table size (in bytes) or the number of RIDs that one RID block can hold. For ANDing multiple indexes, it is 25 percent of the table size.
  - The average is the total value of this field divided by the number of indexes (database/page set combinations).

- **RIDS OBTAINED**
  - The number of RIDs obtained from an index.
  - The average is the total value of this field divided by the number of indexes (database/page set combinations).

- **RIDS EXCEEDED LIMIT**
  - The number of RIDs which exceeded the maximum limit.
  - The average is the total value of this field divided by the number of indexes (database/page set combinations).
Scan Activity

This block shows the total scan activity for each object, performed by the event.

The database name and page set name for each scan are printed if they are available. These do not usually occur in DB2 trace records. The decimal database ID (DBID) and object ID (OBID) occur instead. When possible, OMEGAMON XE for DB2 PE translates the DBID and OBID into database names and page set names. If the translation does not work, the DBID or OBID decimal number is printed instead.

Scan Activity Workload Block Example

Here is an example of the Scan Activity Workload Block.

<table>
<thead>
<tr>
<th>DATABASE</th>
<th>PAGESET</th>
<th>TYPE</th>
<th>SCANS</th>
<th>ROWS PROCESS</th>
<th>QUALIFIED AT</th>
<th>STAGE 1</th>
<th>STAGE 2</th>
<th>INSERTS</th>
<th>UPDATES</th>
<th>DELETES</th>
<th>SCANS</th>
<th>DELETES</th>
<th>SCANS</th>
<th>DELETES</th>
</tr>
</thead>
<tbody>
<tr>
<td>DAPSTEST</td>
<td>SAPSCL</td>
<td>SEQD</td>
<td>63792</td>
<td>63792</td>
<td>63792</td>
<td>33</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2942</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>N/P</td>
<td>6</td>
<td>33</td>
<td>33</td>
<td>0</td>
<td>33</td>
<td>33</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>56</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>DAPSTEST</td>
<td>INDX</td>
<td>SEQD</td>
<td>33</td>
<td>33</td>
<td>33</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>32</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>N/P</td>
<td>6</td>
<td>33</td>
<td>33</td>
<td>33</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>309</td>
<td>2</td>
<td>SEQU</td>
<td>127584</td>
<td>127584</td>
<td>127584</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>5084</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>N/P</td>
<td>4</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>309</td>
<td>INDX</td>
<td>SEQD</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>N/P</td>
<td>6</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TOTAL</td>
<td></td>
<td></td>
<td>73</td>
<td>191446</td>
<td>191411</td>
<td>191411</td>
<td>70</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>8920</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

Field description

Here is a description of the field labels shown in the Scan Activity Workload Block.

DATABASE
The database name.
If the name is not available, the decimal DBID is printed.
If IFCID 058 is not present, DBID information is not available and therefore the field is left blank.

MEMBER
The name of the DB2 member within the DB2 data sharing group. This field shows N/P in a non-data-sharing environment.

PAGESET
The page set name.

Note: If the value shown in the TYPE column is INDX, this column shows the index name if provided by DB2.

TYPE
Indicates whether the scan performed by the data manager is an index file (INDX), a sequential data file (SEQD), or a sequential work file (SEQW).

SCANS
The total number of scans performed by the data manager.

ROWS PROCESS
Number of rows of all record types processed by a scan. As an example:

SELECT A1 FROM TABLE_A WHERE A1=3

Where the table space that contains TABLE_A also contains TABLE_B and TABLE_C. Note that this does not include partitioned table spaces because a partitioned table space can have only one table.
SQL Activity - Scan

For a simple table space, this is a count of all scanned rows from all three tables.

This field is identical to ROWS EXAMINE when the table space is segmented, or when the scan is an index scan.

ROWS EXAMINE
The number of rows of a specific record type processed by the scan. If the table space contains more than one table, scanned rows from the specific table only are counted.

For index scans, this value represents the number of index entries processed.

For a table space containing only one table, the value of ROWS EXAMINE is the same as the value of ROWS PROCESS.

QUALIFIED AT STAGE 1
The total number of rows that were qualified at stage 1.

QUALIFIED AT STAGE 2
The total number of rows that were qualified at stage 2. The value in this field cannot be greater than the value in QUALIFIED AT STAGE 1.

ROWS INSERTS
The number of rows inserted by the data manager.

ROWS UPDATES
The number of rows updated by the data manager.

ROWS DELETES
The number of rows deleted by the data manager.

PAGES SCANNED
The number of Getpage requests the data manager issued to the buffer manager. For an index scan, the field shows the number of Getpage requests on index pages (not index subpages).

RI SCANS
The number of additional Getpage requests the data manager issued to the buffer manager to enforce referential constraints.

RI DELETES
The number of additional rows deleted or set to null due to referential integrity.
Sort Activity

This block shows sort activity for each sort performed by the event.

The sort activity is divided into the following sections:

- "Sort Activity - QW0095/96" on page 46-38, which describes the sort activity recorded by IFCID 095 and 096 (Sort Start and Sort End) pairs
- "Sort Activity - QW0028" on page 46-41, which describes the multiple distinct sort activity recorded by IFCID 028

This topic shows detailed information about “COMPO - Sort Activity - QW0095/96”.

This topic shows detailed information about “COMPO - Sort Activity - QW0028”.

Chapter 46. SQL Activity Report and Trace Blocks 46-37
Sort Activity - QW0095/96

This topic shows detailed information about “COMPO - Sort Activity - QW0095/96”.

The field labels shown in the following sample layout of “COMPO - Sort Activity - QW0095/96” are described in the following section.

MEMBER

The member name of this DB2.
N/A means this DB2 is not part of a data sharing group.

Install parameter MEMBER NAME on panel DSNTIPK, or ZPARM MEMBNAME in DSN6GRP.

Field Name: QWPAMBRN

MAX RETURN CODE

The sort return code:

0 Successful
4 Empty - sort successful
8 Resource unavailable
12 Sort key too long
16 Error detected by fetch routine during input phase
20 Serious processing error

Field Name: QW0096RC

WORKFILES

The number of work files used for both input and merge phases.

Field Name: QW0096WF

RECORDS

The number of records sorted.

Field Name: QW0096NR

TOTAL SortS

The total number of sorts that occurred during the reporting period.

Field Name: SQLTOTSO

INITIAL WORKFILES

The number of initial work files. The sorting of records can take more than one work file. The number of work files needed depends on the distribution of sort key values. The maximum number of work files is limited by the buffer pool size.

Field Name: QW0096IR

RECORD SIZE

46-38 OMEGAMON XE for DB2 PE & PM: Report Reference
The sort record size in bytes (the sort key size and the data area size).

Field Name: QW0096WR

SORT TYPE
The type of sort that occurred. The possible values are:

- **ESA**: ORDER BY sort using the ESA sort hardware instructions
- **ESAG**: GROUP BY sort using the ESA sort hardware instructions
- **ESAT**: ESA tag sort using the ESA sort hardware instructions
- **RCYC**: GROUP RECYCLING sort using the ESA sort hardware instructions
- **REG**: Regular sort
- **NONE**: No sort occurred.

Field Name: QW0096TS

SORT KEYS
The number of sort keys.

Field Name: QW0096SK

SORT COLUMNS
The number of sort columns.

Field Name: QW0096SC

KEY SIZE
The sort key size in bytes.

Field Name: QW0096KL

MERGE PASSES
The number of merge passes during sort processing.

Field Name: QW0096MP

AET/SORT
The average elapsed time per sort.

Field Name: SQLAVTSO

DATA SIZE
The sort data area size in bytes.

Field Name: QW0096DL

ROWS DELETED
The number of rows deleted because records were merged for the evaluation of column functions with GROUP BY.

Field Name: QW0096RL

PARTITIONING
Indicates whether the sorted records were partitioned.

Field Name: QW0096PP

PARTITION TYPE
Indicates when partitioning took place:

- **W** The work file was partitioned at the end of the input phase. No merge occurred.
- **M** The output was partitioned during the last merge pass.
- **O** One record was put into one partition.
- **P** The records were presorted before being partitioned.
- **N** The work file was not partitioned.

**Field Name:** QW0096PT

**W-FILES PART**

The number of work files, equal to the degree of parallelism, that sort has partitioned.

**Field Name:** QW0096PW

**PARTIT & SORTING**

Indicates whether the input records were only partitioned or partitioned and sorted:

- **YES** The records were only partitioned.
- **NO** The records were partitioned and sorted.

**Field Name:** QW0096PO
Sort Activity - QW0028

This topic shows detailed information about “COMPO - Sort Activity - QW0028”.

Sort Activity - QW0028

The field labels shown in the following sample layout of “COMPO - Sort Activity - QW0028” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NO. OF WORKFILES</td>
<td>87</td>
</tr>
<tr>
<td>TOTAL MULTIPLE DS</td>
<td>152</td>
</tr>
<tr>
<td>MULTIPLE_DISTINCT SORTS</td>
<td>277</td>
</tr>
<tr>
<td>REQUESTED WORKFILES</td>
<td>767</td>
</tr>
<tr>
<td>ACTUAL WORKFILES</td>
<td>613</td>
</tr>
<tr>
<td>RECORDS IN WORKFILE</td>
<td>221</td>
</tr>
<tr>
<td>PARTITION PARALLEL</td>
<td>1090</td>
</tr>
<tr>
<td>RECORDS LAST MERGE</td>
<td>612</td>
</tr>
<tr>
<td>PARTITIONING ONE RECORD</td>
<td>571</td>
</tr>
<tr>
<td>PRE-SORTED RECORDS</td>
<td>134</td>
</tr>
<tr>
<td>RECORDS MDS GROUPBY</td>
<td>22</td>
</tr>
<tr>
<td>CURRENT MERGE PASS</td>
<td>88</td>
</tr>
<tr>
<td>TOTAL MDS GROUPS</td>
<td>132</td>
</tr>
<tr>
<td>RECORDS MDS GROUPS</td>
<td>44</td>
</tr>
<tr>
<td>MAX REQUESTED</td>
<td>767</td>
</tr>
<tr>
<td>AVG REQUESTED</td>
<td>767.00</td>
</tr>
<tr>
<td>MAX NOT ACQUIRED</td>
<td>154</td>
</tr>
<tr>
<td>AVG NOT ACQUIRED</td>
<td>154.00</td>
</tr>
</tbody>
</table>

**NO. OF WORKFILES**

The number of records sorted into work files after the sort input phase.

Field Name: QW0028NR

**TOTAL MULTIPLE DS**

Total number of multiple distinct sorts.

Field Name: QW0028DS

**MULTIPLE_DISTINCT SORTS**

The multiple distinct sort currently being processed.

Field Name: QW0028DC

**REQUESTED WORKFILES**

The number of work files requested from the buffer manager at the beginning of each merge pass (MVS/ESA 3.1.3). It is valid if TYPE equals S.

If this field is greater than WORKFILES ACQ, there is another merge pass. If both fields are equal, this is the last or only merge pass.

Field Name: QW0028WA

**ACTUAL WORKFILES**

The number of work files actually acquired from the buffer manager at the beginning of each merge pass (MVS/ESA 3.1.3). It is valid if TYPE equals S.

Field Name: QW0028WG

**RECORDS IN WORKFILE**

The number of records in the work file during work file partitioning.

Field Name: SQ28TYZ

**PARTITION PARALLEL**

The partition work file number. The value in this field is 0 if partitioning is not requested. If partitioning is requested, the value can be from 1 to n, where n is the degree of parallelism. It is valid if TYPE equals Z, W, X, K, M, L, T, O, U, V, P, or Y.

Field Name: QW0028PW

**RECORDS LAST MERGE**

The number of sort records in the partition work file during the last merge.
Sort Activity - QW0028

Field Name: SQ28TYQ
PARTITIONING ONE RECORD
The number of times that partitioning occurred when only one record was sorted and put into a partition work file.

Field Name: SQ28TYT
PRE-SORTED RECORDS
The number of times partitioning occurred when presorted records are put into multiple work files.

Field Name: SQ28TYV
RECORDS MDS GROUPS
The number multiple distinct loops containing a number of multiple distinct sorts.

Field Name: SQ28TYB
CURRENT MERGE PASS
The current merge pass. It is issued at the end of the merge pass and, therefore, valid if TYPE equals E.

Field Name: QW0028MP
TOTAL MDS GROUPS
The total number of multiple distinct sort groups.

Field Name: QW0028DG
RECORDS MDS GROUPBY
The number of records read into a group at the start of the GROUPBY phase for a multiple distinct sort.

Field Name: SQ28TYD
MAX REQUESTED
The maximum number of work files requested from buffer manager during merge passes.

Field Name: SQ28MAXR
AVG REQUESTED
The average number of work files requested from buffer manager during merge passes.

Field Name: SQ28AVRQ
MAX NOT ACQUIRED
The maximum number of work files requested but not received from buffer manager during merge passes.

Field Name: SQ28MXNA
AVG NOT ACQUIRED
The average of work files requested but not received from buffer manager during merge passes.

Field Name: SQ28AVNA
Host Variables

This block shows the host variables to represent the values that will be sent to or received from the SQL statement.

Note:
1. The number of shown host variables is limited to 100.
2. A warning message is issued in DPMLOG, if the limit is reached or if SQLDA entries are missing.

Workload Host Variables Example

Here is an example of the workload host variables.

```--- HOST VARIABLES -------------------------------------------------------------------------------------------------------
LOCATION : SYSDSNI COLLID : PMDEMO PROGNAME : DYNSEL08 CONSIS_TOKEN : X'5A427634E644C54'
STMT_NO : 133 FORMAT : 1 = COMPLETE NO.SQLDA ENTRIES : 3
..........................................................................................................................
ENTRY_NO. : 1 NAME : NULL_INDICATOR : NO SQLTYPE : 452
DATA_TYPE : FIXED-LENGTH CHARACTER STRING DATA_LENGTH : 6
PRECISION : N/A SCALE : N/A ADDR_HOST_VAR : X'00064DD0' ADDR_IND_VAR : X'000650A0'
DATA : 000001
........................................................................................................................
ENTRY_NO. : 2 NAME : NULL_INDICATOR : NO SQLTYPE : 452
DATA_TYPE : FIXED-LENGTH CHARACTER STRING DATA_LENGTH : 6
PRECISION : N/A SCALE : N/A ADDR_HOST_VAR : X'00064DD8' ADDR_IND_VAR : X'000650A0'
DATA : 000001
........................................................................................................................
ENTRY_NO. : 3 NAME : NULL_INDICATOR : NO SQLTYPE : 452
DATA_TYPE : FIXED-LENGTH CHARACTER STRING DATA_LENGTH : 6
PRECISION : N/A SCALE : N/A ADDR_HOST_VAR : X'00064DE0' ADDR_IND_VAR : X'000650A0'
DATA : 000001
........................................................................................................................
```

Field description

Here is a description of the field labels shown in the workload host variables block.

LOCATION

Location name.

Field Name: QW0247LN

COLLID

Package collection identifier.

Field Name: QW0247PC

PROGNAME

Program name.

Field Name: QW0247PN

CONSIS_TOKEN

Not present (N/P) is shown for this field if the value is X’40’ or X’00’; otherwise the hexadecimal value of the field is shown.

Field Name: QW0247TS

STMT_NO

Statement number.

Field Name: QW0247SN

FORMAT
SQL Activity - Host Variables

The format of the SQLDA. Possible values are:

B'1000'

0 - COMPRESSED
Is a compressed form of the SQLDA.

B'0100'

1 - COMPLETE
Is a complete SQLDA containing the data type, address,
and address of the indicator variable for each host variable.

B'0010'

2 - FIXED LENGTH
Is a variable length character format containing the length
of the string and text.

? - UNKNOWN
Is shown, if none of the above field names is used.

Field Name: QW0247FE
ENTRY_NO.
SQLDA entry number.
Field Name: QW0247NO
NAME
SQLDA name, if Format 1 SQLDA. The first two bytes are the length of the
NAME and are not shown.
Field Name: QW0247NA
NULL_INDICATOR
Null indicator values:
• YES, if X'00'
• NO, if X'FF'
Field Name: QW0247NL
SQLTYPE
SQL type (see DB2 SQL Reference).
Field Name: QW0247TY
DATA TYPE
DATA TYPE is derived as described in DB2 SQL Reference, based on the
SQLTYPE:
384, 385
DATE
388, 389
TIME
392, 393
TIMESTAMP
448, 449
VARYING LENGTH CHARACTER STRING
SQL Activity - Host Variables

452, 453
FIXED-LENGTH CHARACTER STRING

456, 457
LONG VARYING CHARACTER STRING

480, 481
FLOATING POINT

484, 485
PACKED DECIMAL

496, 497
LARGE INTEGER

500, 501
SMALL INTEGER

Note:
1. Any other SQLTYPES are shown as: NON DISPLAYABLE DATA
2. Values are shown in DB2 internal format. For details refer to the DB2 information center (http://publib.boulder.ibm.com/infocenter/dzichelp/v2r2/index.jsp).

Field Name: QW0247TY

PRECISION
If the field is decimal (484 or 485), this is the precision.

Field Name: QW0247LP

SCALE
If the field is decimal (484 or 485), this is the scale.

Field Name: QW0247LS

ADDR_HOST_VAR
The address of the host variable in the application address space.

Field Name: QW0247PT

ADDR_IND_VAR
The address of the indicator variable, if the value in QW0247TY is odd (NULLABLE).

Field Name: QW0247IN

DATA
The host variable data.

Field Name: QW0247DA
**Workload Highlight**

This block shows the highlights of the workload activity performed by the event. All workload fields available in the SORTBY option are included.

**Workload Highlights (HILITE) Block Example**

Here is an example of the workload highlight (HILITE) block.

```
--- WORKLOAD HILITE
Scans : 8  Recs/Sort : 3.00  I/O REQS : 1  Suspends : 2  Exits : 2  AMS : 1
RowSPROC : 8  Work/Sort : 2.00  AET/I/O : 1.374752  AET/Suspend : 0.485483  AET/Exit : 0.048234  AET/AMS : 0.094745
Pagescan : 47  Pass/Sort : 2.00  DataC apt : YES  RIDS Unused : 2  Checkcon : REJECTED  Degree Reduction : 3
LOB_Pagescan : 12345  LOB_UPD_PAGE : 12345
```

**Field description**

Here is a description of the field labels shown in the workload highlight (HILITE) block.

**SCANS**
- The total number of scans performed by the data manager.

**RECS/SORT**
- The average number of records per sort.

**I/O REQS**
- The number of SYNCHRONOUS and ASYNCHRONOUS READ and SYNCHRONOUS WRITE I/O requests per event.

**SUSPENDS**
- The number of LOCK SUSPENSIONS per event.

**EXITS**
- The number of validation, encode edit, and decode edit exits per event.

**AMS**
- The number of times Access Method Services (AMS) was invoked within the event. AMS can be invoked by:
  - Creating a DB2 page set (table space, table partition, index space)
  - Expanding an existing DB2 page set
  - Deleting a DB2 page set

**ROWSPROC**
- The number of rows processed (of all record types).

**WORK/SORT**
- The average number of work files per sort.

**AET/I/O**
- The average elapsed time I/O requests.

**AET/SUSP**
- The average elapsed time for LOCK SUSPENSIONS.

**AET/EXIT**
- The average elapsed time per EXIT invocation.

**AET/AMS**
- The average elapsed time of the AMS invocations within the event.

**PAGESCAN**
- The number of pages scanned.

**DATACAPT**
- The data capture indicator; shows whether IFCID 188 is present.
SQL Activity - Workload Highlight

RIDS UNUSED
The number of times RID list processing was not used because nor RID storage was available or the number of RIDs exceeded the maximum limit.

CHECKCON
Indicates that a table check constraint was performed for the current SQL event:

OK The check constraint was ok.

REJECTED The row to be inserted or updated was rejected due to a check constraint.

N/P No check was performed.

DEGREE REDUCTION
The difference between planned and negotiated run time degree.

LOB_PAGSCAN
The number of LOB pages scanned.

LOB_UPD_PAGE
The number of LOB pages updated.

SYSPLEX QUERY PARALLELISM USED
This field is shown if the query is executed on more than one member, this field shows the number of members. Otherwise this field is blank.

Note:
1. All fields in the highlight block are printed. If other detail blocks are requested, then some of the highlights are shown twice, once in the highlight block and again in the detail block.
2. If the records required for a field are not present, N/P is printed for that field. N/A is printed if the field is not relevant to the level of DB2.
SQL Activity - Workload Highlight

These topics provide information about the Statistics reports.

This section provides examples of the statistics default layout for SHORT and LONG. Descriptions of the fields in the layout are described in the next section. Because the layout of the report and trace is the same (with the exception of the highlights block), only a report example is reproduced here.

When data from a particular DB2 version is processed, N/A is printed for all fields in the report that are not applicable to that version.

You can use the user-tailored reporting (UTR) facility to modify the layouts and store the changes. If you do this, store your layouts under a different name to avoid confusion and keep the layouts relevant to this documentation.

Note: For an introduction to the Statistics report set and general Statistics information refer to the Reporting User’s Guide. It also provides information on input to Statistics reports.

Chapter 47, “Statistics Short Report,” on page 47-1
This topic introduces the short version of the Statistics reports.

This topic introduces the long version of the Statistics reports.

Chapter 49, “Statistics Report and Trace Blocks,” on page 49-1
Chapter 50, “The Statistics Save-File Utility,” on page 50-1
Use the Save-File utility to migrate and convert Statistics Save data sets into a format suitable for OMEGAMON XE for DB2 PE V5,2.

Chapter 51, “The Statistics File Data Set and Output Records,” on page 51-1
Use the FILE subcommand to format DB2 Statistics records and write them to sequential data sets suitable for use by the DB2 load utility. You can store unreduced Statistics data into the performance database. Use the performance database to produce tailored reports using a reporting facility such as Query Management Facility (QMF).
Chapter 47. Statistics Short Report

This topic introduces the short version of the Statistics reports.

The SHORT layout presents selected data from all Statistics categories using the following blocks of data:

- “Accounting Rollup” on page 49-9
- “Buffer Pool General” on page 49-15
- “Buffer Pool Write” on page 49-31
- “CPU Times” on page 49-41
- “Data Sharing Locking” on page 49-47
- “EDM Pool Activity (DB2 9)” on page 49-87
- “EDM Pool Activity (DB2 10 or later)” on page 49-93
- “Global DDF Activity” on page 49-98
- “Group Buffer Pool Activity” on page 49-103
- “Highlights” on page 49-115
- “Locking Activity” on page 49-133
- “Log Activity” on page 49-137
- “Miscellaneous” on page 49-142
- “Open/Close Activity” on page 49-146
- “Plan/Package Activity” on page 49-148
- “Query Parallelism” on page 49-153
- “RID List Processing” on page 49-162
- “ROWID” on page 49-166
- “SQL DCL” on page 49-177
- “SQL DDL” on page 49-179
- “SQL DML” on page 49-185
- “Stored Procedures” on page 49-168
- “Subsystem Services” on page 49-169
- “Triggers” on page 49-187
- “Use Currently Committed” on page 49-188
- “User-Defined Functions” on page 49-189
- “Workfile Database” on page 49-190

Use the following command to print a Statistics short report:

```sql
:STATISTICS
REPORT
LAYOUT (SHORT)
:;```
Chapter 48. Statistics Long Report

This topic introduces the long version of the Statistics reports.

The following report is an example of a member-scope statistics long report, produced with the command:

```
STATISTICS REPORT DSETSTAT LAYOUT(LONG)
```

Statistics Long Report

This is an example of a long report for Statistics.

<table>
<thead>
<tr>
<th>SQL DML</th>
<th>QUANTITY</th>
<th>/SECOND</th>
<th>/THREAD</th>
<th>/COMMIT</th>
<th>SQL DCL</th>
<th>QUANTITY</th>
<th>/SECOND</th>
<th>/THREAD</th>
<th>/COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECT</td>
<td>3529.00</td>
<td>0.17</td>
<td>18.97</td>
<td>0.14</td>
<td>LOCK TABLE</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>INSERT</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>GRANT</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>NUMBER OF ROWS</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>REVOKE</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>UPDATE</td>
<td>3645.00</td>
<td>0.17</td>
<td>19.60</td>
<td>0.14</td>
<td>SET HOST VARIABLE</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>NUMBER OF ROWS</td>
<td>116.00</td>
<td>0.01</td>
<td>0.62</td>
<td>0.00</td>
<td>SET CURRENT SSQL</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>MERGE</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>SET CURRENT DEGREE</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DELETE</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>SET CURRENT RULES</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>NUMBER OF ROWS</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>SET CURRENT PATH</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>PREPARE</td>
<td>696.00</td>
<td>0.03</td>
<td>3.74</td>
<td>0.03</td>
<td>SET CURRENT PRECISION</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DESCRIBE</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>SET CURRENT TABLE</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DESCRIBE TABLE</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>SET CURRENT ID</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>OPEN</td>
<td>18225.00</td>
<td>0.86</td>
<td>97.98</td>
<td>0.72</td>
<td>SET CONNECTION</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>CLOSE</td>
<td>18225.00</td>
<td>0.86</td>
<td>97.98</td>
<td>0.72</td>
<td>SET CURRENT RULES</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>FETCH</td>
<td>18225.00</td>
<td>0.86</td>
<td>97.98</td>
<td>0.72</td>
<td>SET CURRENT PATH</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>NUMBER OF ROWS</td>
<td>580.00</td>
<td>0.03</td>
<td>3.12</td>
<td>0.02</td>
<td>SET CURRENT RULES</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>TOTAL DML</td>
<td>62545.00</td>
<td>2.94</td>
<td>336.26</td>
<td>2.46</td>
<td>SET CURRENT ROWS</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>STORED PROCEDURES</th>
<th>QUANTITY</th>
<th>/SECOND</th>
<th>/THREAD</th>
<th>/COMMIT</th>
<th>TRIGGERS</th>
<th>QUANTITY</th>
<th>/SECOND</th>
<th>/THREAD</th>
<th>/COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>CALL STATEMENT EXECUTED</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>STATEMENT TRIGGER ACTIVATED</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>PROCEDURE ABENDED</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>ROW TRIGGER ACTIVATED</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>CALL STATEMENT TIMED OUT</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>SQL ERROR OCCURRED</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>CALL STATEMENT REJECTED</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>USER DEFINED FUNCTIONS</th>
<th>QUANTITY</th>
<th>/SECOND</th>
<th>/THREAD</th>
<th>/COMMIT</th>
<th>ROW ID</th>
<th>QUANTITY</th>
<th>/SECOND</th>
<th>/THREAD</th>
<th>/COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXECUTED</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>DIRECT ACCESS</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>ABENDED</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>INDEX USED</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>TIMED OUT</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>TABLE SPACE SCAN USED</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>REJECTED</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

© Copyright IBM Corp. 1985, 2013
### HIGHLIGHTS

Interval Start: 10/14/13 21:54:00.00
SAMPLING Start: 10/14/13 21:54:00.00
Total Threads: 186.00

Interval End: 10/15/13 03:49:00.00
SAMPLING End: 10/15/13 03:49:00.00
Total Commits: 25399.00

- **Time Elapsed:** 5:54:59.996886
- **Outage Elapsed:** 0.000000
- **Data Sharing Member:** N/A

### SQL DDL

<table>
<thead>
<tr>
<th>SQL DDL</th>
<th>Quantity /Second /Thread /Commit</th>
<th>SQL DDL Continued</th>
<th>Quantity /Second /Thread /Commit</th>
</tr>
</thead>
<tbody>
<tr>
<td>CREATE TABLE</td>
<td>0.00 0.00 0.00 0.00</td>
<td>DROP TABLE</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>CREATE GLOBAL TEMP TABLE</td>
<td>0.00 0.00 0.00 0.00</td>
<td>DROP INDEX</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>DECLARE GLOBAL TEMP TABLE</td>
<td>0.00 0.00 0.00 0.00</td>
<td>DROP VIEW</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>CREATE AUXILIARY TABLE</td>
<td>0.00 0.00 0.00 0.00</td>
<td>DROP SYNONYM</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>CREATE INDEX</td>
<td>0.00 0.00 0.00 0.00</td>
<td>DROP TABLESPACE</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>CREATE VIEW</td>
<td>0.00 0.00 0.00 0.00</td>
<td>DROP DATABASE</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>CREATE SYNTHYM</td>
<td>0.00 0.00 0.00 0.00</td>
<td>DROP STOGROUP</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>CREATE TABLESPACE</td>
<td>0.00 0.00 0.00 0.00</td>
<td>DROP ALIAS</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>CREATE DATABASE</td>
<td>0.00 0.00 0.00 0.00</td>
<td>DROP PACKAGE</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>CREATE STOGROUP</td>
<td>0.00 0.00 0.00 0.00</td>
<td>DROP DISTINCT TYPE</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>CREATE ALIAS</td>
<td>0.00 0.00 0.00 0.00</td>
<td>DROP FUNCTION</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>CREATE DISTINCT TYPE</td>
<td>0.00 0.00 0.00 0.00</td>
<td>DROP PROCEDURE</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>CREATE FUNCTION</td>
<td>0.00 0.00 0.00 0.00</td>
<td>DROP TRIGGER</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>CREATE PROCEDURE</td>
<td>0.00 0.00 0.00 0.00</td>
<td>DROP SEQUENCE</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>CREATE TRIGGER</td>
<td>0.00 0.00 0.00 0.00</td>
<td>DROP ROLE</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>CREATE SEQUENCE</td>
<td>0.00 0.00 0.00 0.00</td>
<td>DROP TRUSTED CONTEXT</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>CREATE TRUSTED CONTEXT</td>
<td>0.00 0.00 0.00 0.00</td>
<td>DROP VARIABLE</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>CREATE MASK / PERMISSION</td>
<td>0.00 0.00 0.00 0.00</td>
<td>RENAME TABLE</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>CREATE VARIABLE</td>
<td>0.00 0.00 0.00 0.00</td>
<td>RENAME INDEX</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>ALTER TABLE</td>
<td>0.00 0.00 0.00 0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALTER INDEX</td>
<td>0.00 0.00 0.00 0.00</td>
<td>TRUNCATE TABLE</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>ALTER VIEW</td>
<td>0.00 0.00 0.00 0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALTER TABLESPACE</td>
<td>0.00 0.00 0.00 0.00</td>
<td>COMMENT ON</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>ALTER DATABASE</td>
<td>0.00 0.00 0.00 0.00</td>
<td>LABEL ON</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>ALTER STOGROUP</td>
<td>0.00 0.00 0.00 0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALTER FUNCTION</td>
<td>0.00 0.00 0.00 0.00</td>
<td>TOTAL</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>ALTER PROCEDURE</td>
<td>0.00 0.00 0.00 0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALTER SEQUENCE</td>
<td>0.00 0.00 0.00 0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALTER TRIGGER</td>
<td>0.00 0.00 0.00 0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALTER TRUSTED CONTEXT</td>
<td>0.00 0.00 0.00 0.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALTER MASK / PERMISSION</td>
<td>0.00 0.00 0.00 0.00</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### EDM Pool

<table>
<thead>
<tr>
<th></th>
<th>Quantity /Second /Thread /Commit</th>
</tr>
</thead>
<tbody>
<tr>
<td>PAGES IN EDM POOL (ABOVE)</td>
<td>25600.00 N/A N/A N/A</td>
</tr>
<tr>
<td>HELD BY EDM</td>
<td>143.00 N/A N/A N/A</td>
</tr>
<tr>
<td>STEALABLE PAGES</td>
<td>49.00 N/A N/A N/A</td>
</tr>
<tr>
<td>FREE PAGES</td>
<td>25457.00 N/A N/A N/A</td>
</tr>
<tr>
<td>% PAGES IN USE</td>
<td>0.37 N/A N/A N/A</td>
</tr>
<tr>
<td>FAILS DUE TO EDM POOL FULL</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>PAGES IN STMT POOL (ABOVE)</td>
<td>28346.00 N/A N/A N/A</td>
</tr>
<tr>
<td>HELD BY STATEMENTS</td>
<td>184.00 N/A N/A N/A</td>
</tr>
<tr>
<td>FREE PAGES</td>
<td>28162.00 N/A N/A N/A</td>
</tr>
<tr>
<td>% PAGES IN USE</td>
<td>0.00 N/A N/A N/A</td>
</tr>
<tr>
<td>FAILS DUE TO STMT POOL FULL</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>PAGES IN SKEL POOL (ABOVE)</td>
<td>25600.00 N/A N/A N/A</td>
</tr>
<tr>
<td>HELD BY SKCT</td>
<td>6.00 N/A N/A N/A</td>
</tr>
<tr>
<td>HELD BY SKPT</td>
<td>110.00 N/A N/A N/A</td>
</tr>
<tr>
<td>STEALABLE PAGES</td>
<td>216.00 N/A N/A N/A</td>
</tr>
<tr>
<td>FREE PAGES</td>
<td>25484.00 N/A N/A N/A</td>
</tr>
<tr>
<td>% PAGES IN USE</td>
<td>0.00 N/A N/A N/A</td>
</tr>
<tr>
<td>FAILS DUE TO SKEL POOL FULL</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>DBD REQUESTS</td>
<td>22022.00 1.03 118.40 0.87</td>
</tr>
<tr>
<td>DBD NOT FOUND</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>DBD HIT RATIO (%)</td>
<td>100.00 N/A N/A N/A</td>
</tr>
<tr>
<td>CT REQUESTS</td>
<td>186.00 0.01 1.00 0.01</td>
</tr>
<tr>
<td>CT NOT FOUND</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>CT HIT RATIO (%)</td>
<td>100.00 N/A N/A N/A</td>
</tr>
<tr>
<td>PT REQUESTS</td>
<td>43044.00 2.02 231.42 1.69</td>
</tr>
<tr>
<td>PT NOT FOUND</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>PT HIT RATIO (%)</td>
<td>100.00 N/A N/A N/A</td>
</tr>
</tbody>
</table>

48-2 OMEGAMON XE for DB2 PE & PM: Report Reference
### Dynamic SQL Stmt

<table>
<thead>
<tr>
<th>Quantity</th>
<th>/Second</th>
<th>/Thread</th>
<th>/Commit</th>
</tr>
</thead>
<tbody>
<tr>
<td>FULL PREPARES</td>
<td>696.00</td>
<td>0.03</td>
<td>3.74</td>
</tr>
<tr>
<td>SHORT PREPARES</td>
<td>696.00</td>
<td>0.03</td>
<td>3.74</td>
</tr>
<tr>
<td>MODIFIED STATEMENTS</td>
<td>59.00</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>

### Subsystem Services

<table>
<thead>
<tr>
<th>Quantity</th>
<th>/Second</th>
<th>/Thread</th>
<th>/Commit</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMIT PHASE 1</td>
<td>116.00</td>
<td>0.01</td>
<td>1.00</td>
</tr>
<tr>
<td>COMMIT PHASE 2</td>
<td>7174.00</td>
<td>0.34</td>
<td>38.57</td>
</tr>
<tr>
<td>TERMINATE</td>
<td>418.00</td>
<td>0.02</td>
<td>2.25</td>
</tr>
</tbody>
</table>

### Open/Close Activity

<table>
<thead>
<tr>
<th>Quantity</th>
<th>/Second</th>
<th>/Thread</th>
<th>/Commit</th>
</tr>
</thead>
<tbody>
<tr>
<td>TAPE VOLUME CONTENTION WAIT</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>READ DELAYED-UNAVAIL.RESOURCE</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>
| ARCHIVE LOG WRITE
dead allocation | 0.00 | 0.00 | 0.00 | 0.00 |
| ARCHIVE LOG WRITE
dead allocation | 0.00 | 0.00 | 0.00 | 0.00 |
| UNAVAIL.OUTPUT LOG BUFFER | 0.00 | 0.00 | 0.00 | 0.00 |
| LOG RECORDS CREATED | 1741.00 | 0.08 | 9.36 | 0.07 |
| LOG CI CREATED | 191.00 | 0.01 | 1.03 | 0.01 |
| LOG CI C乾隆 (LOG1/2) | 5743.00 | 0.27 | 30.88 | 0.23 |
| LOG CI C乾隆 (LOG1/2) | 5743.00 | 0.27 | 30.88 | 0.23 |
| LOG RATE FOR 1 LOG (MB) | N/A | 0.00 | N/A | N/A |

### Plan/Packaging Processing

<table>
<thead>
<tr>
<th>Quantity</th>
<th>/Second</th>
<th>/Thread</th>
<th>/Commit</th>
</tr>
</thead>
<tbody>
<tr>
<td>PLAN ALLOCATION ATTEMPTS</td>
<td>186.00</td>
<td>0.01</td>
<td>1.00</td>
</tr>
<tr>
<td>PLAN ALLOCATION SUCCESSFUL</td>
<td>186.00</td>
<td>0.01</td>
<td>1.00</td>
</tr>
<tr>
<td>PACKAGE ALLOCATION ATTEMPT</td>
<td>18341.00</td>
<td>0.86</td>
<td>98.61</td>
</tr>
<tr>
<td>PACKAGE ALLOCATION SUCCESS</td>
<td>18341.00</td>
<td>0.86</td>
<td>98.61</td>
</tr>
<tr>
<td>PLANS BOUND</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>BIND ADD SUBCOMMANDS</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>BIND REPLACE SUBCOMMANDS</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>BASE BOUND NO PLAN-ID</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>PACKAGES BOUND</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>BIND ADD PACKAGE SUBCOMMAND</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>
BIND REPLACE PACKAGE SUBCOMMAND 0.00 0.00 0.00 0.00
AUTOMATIC BIND ATTEMPTS 0.00 0.00 0.00 0.00
AUTOMATIC BINDS SUCCESSFUL 0.00 0.00 0.00 0.00
AUTO.BIND INVALID RES. IDS 0.00 0.00 0.00 0.00
AUTO.BIND PACKAGE ATTEMPTS 0.00 0.00 0.00 0.00
AUTO.BIND PACKAGES SUCCESS 0.00 0.00 0.00 0.00
REBIND SUBCOMMANDS 0.00 0.00 0.00 0.00
ATTEMPTS TO REBIND A PLAN 0.00 0.00 0.00 0.00
PLANS REBOUND 0.00 0.00 0.00 0.00
REBIND PACKAGE SUBCOMMANDS 0.00 0.00 0.00 0.00
ATTEMPTS TO REBIND PACKAGE 0.00 0.00 0.00 0.00
PACKAGES REBOUND 0.00 0.00 0.00 0.00
FREE PLAN SUBCOMMANDS 0.00 0.00 0.00 0.00
ATTEMPTS TO FREE A PLAN 0.00 0.00 0.00 0.00
PLANS FREED 0.00 0.00 0.00 0.00
FREE PACKAGE SUBCOMMANDS 0.00 0.00 0.00 0.00
ATTEMPTS TO FREE A PACKAGE 0.00 0.00 0.00 0.00
PACKAGES FREED 0.00 0.00 0.00 0.00

LOCATION: OMPDBE3
GROUP: DBE3
SUBSYSTEM: SE31
INTERVAL FROM: 10/14/13 21:54:00.00
DB2 VERSION: V11
SCOPE: MEMBER
INTERVAL TO: 10/15/13 03:49:00.00

--- HIGHLIGHTS
INTERVAL START : 10/14/13 21:54:00.00
INTERVAL END : 10/15/13 03:49:00.00
TOTAL THREADS : 186.00
TOTAL COMMITS : 25399.00
INTERVAL ELAPSED: 5:54:59.996886
OUTAGE ELAPSED: 0.000000
DATA SHARING MEMBER: N/A

RID LIST PROCESSING QUANTITY /SECOND /THREAD /COMMIT
SUCCESSFUL 0.00 0.00 0.00 0.00
TOTAL AUTH ATTEMPTS 256.00 0.01 1.38 0.01
TOTAL AUTH SUCC 256.00 0.01 1.38 0.01
NOT USED-NO STORAGE 0.00 0.00 0.00 0.00

LOCATION: OMPDBE3
GROUP: DBE3
SUBSYSTEM: SE31
INTERVAL FROM: 10/14/13 21:54:00.00
DB2 VERSION: V11
SCOPE: MEMBER
INTERVAL TO: 10/15/13 03:49:00.00

--- HIGHLIGHTS
INTERVAL START : 10/14/13 21:54:00.00
INTERVAL END : 10/15/13 03:49:00.00
TOTAL THREADS : 186.00
TOTAL COMMITS : 25399.00
INTERVAL ELAPSED: 5:54:59.996886
OUTAGE ELAPSED: 0.000000
DATA SHARING MEMBER: N/A
ACC QU INACT CONNS (TYPE 2) 0.00 0.00 N/A N/A N/C
CUR QU INACT CONNS (TYPE 2) 0.00 0.00 N/A N/A N/C
MIN QUEUE TIME 0.000000 N/A N/A N/A N/C
MAX QUEUE TIME 0.000000 N/A N/A N/A N/C
AVG QUEUE TIME 0.000000 N/A N/A N/A N/C
HWM QU INACT CONNS (TYPE 2) 0.00 N/A N/A N/A N/C
CUR ACTIVE AND DISCON DBATS 0.00 N/A N/A N/A N/C
HWM ACTIVE AND DISCON DBATS 0.00 N/A N/A N/A N/C
HWM TOTL REMOTE CONNECTIONS 0.00 N/A N/A N/A N/C
CUR DISCON DBATS NOT IN USE 0.00 N/A N/A N/A N/C
HWM DISCON DBATS NOT IN USE 0.00 N/A N/A N/A N/C
DBATS CREATED 0.00 N/A N/A N/A N/C
DISCON (POOL) DBATS REUSED 0.00 N/A N/A N/A N/C
CUR ACTIVE DBATS-BND DEALLC 0.00 N/A N/A N/A N/C
HWM ACTIVE DBATS-BND DEALLC 0.00 N/A N/A N/A N/C

LOCATION: OMPDBE3 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0)
GROUP: DBE3 STATISTICS REPORT - LONG
MEMBER: SE31 SUBSYSTEM: SE31

INTERVAL START : 10/14/13 21:54:00.00 SAMPLING START: 10/14/13 21:54:00.00 TOTAL THREADS : 186.00
INTERVAL END : 10/15/13 03:49:00.00 SAMPLING END : 10/15/13 03:49:00.00 TOTAL COMMITS : 25399.00
INTERVAL ELAPSED: 5:54:59.996886 OUTAGE ELAPSED: 0.000000 DATA SHARING MEMBER: N/A

CPU TIMES
TCB TIME PREEMPT SRB NONPREEMPT SRB CP CPU TIME PREEMPT IIP SRB CP CPU /COMMIT
------------------------------- --------------- --------------- --------------- --------------- --------------- --------------
SYSTEM SERVICES ADDRESS SPACE 10.612149 0.105322 1.472960 12.190431 0.228652 0.000480
DATABASE SERVICES ADDRESS SPACE 0.185496 0.460905 0.210459 0.856860 0.787446 0.000034
IRLM 0.000763 0.000000 3.407027 3.407790 0.000000 0.000134
DDF ADDRESS SPACE 1.024786 0.001720 0.088041 1.114546 0.000000 0.000044
TOTAL 11.823195 0.567947 5.178486 17.569627 1.016098 0.000692

DB2 APPL.PROGR.INTERFACE QUANTITY /SECOND /THREAD /COMMIT DATA CAPTURE QUANTITY /SECOND /THREAD /COMMIT
--------------------------- -------- ------- ------- ------- --------------------------- -------- ------- ------- -------
ABENDS 0.00 0.00 0.00 0.00 LOG RECORDS CAPTURED 0.00 0.00 0.00 0.00
UNRECOGNIZED 0.00 0.00 0.00 0.00 LOG READS PERFORMED 0.00 0.00 0.00 0.00
COMMAND REQUESTS 70.00 0.00 0.38 0.00 LOG RECORDS RETURNED 0.00 0.00 0.00 0.00
READA REQUESTS 1069.00 0.05 5.75 0.04 DATA ROWS RETURNED 0.00 0.00 0.00 0.00
READS REQUESTS 714.00 0.03 3.84 0.03 DESCRIBES PERFORMED 0.00 0.00 0.00 0.00
WRITE REQUESTS 0.00 0.00 0.00 0.00 DATA DESCRIPTIONS RETURNED 0.00 0.00 0.00 0.00
TOTAL 1853.00 0.09 9.96 0.07

IFC DEST. WRITTEN NOT WRTN BUF.OVER NOT ACCP WRT.FAIL IFC RECORD COUNTS WRITTEN NOT WRTN
--------- -------- -------- -------- -------- ----------------- -------- --------
SMF 3931.00 0.00 0.00 0.00 0.00 SYSTEM RELATED 556.00 0.00
GTF 48-6 0.00 0.00 N/A 0.00 0.00 DATABASE RELATED 556.00 0.00
OP1 381.00 0.00 N/A 0.00 N/A ACCOUNTING 78.00 0.00
OP2 0.00 0.00 N/A 0.00 N/A START TRACE 0.00 0.00
OP3 70.00 0.00 N/A 0.00 N/A STOP TRACE 0.00 0.00
OP4 1091.00 0.00 N/A 0.00 N/A SYSTEM PARAMETERS 1242.00 0.00
OP5 0.00 0.00 N/A 0.00 N/A SYS.PARMS-BPOOLS 556.00 0.00
OP6 0.00 0.00 N/A 0.00 N/A AUDIT 0.00 0.00
OP7 0.00 0.00 N/A 0.00 N/A 5473.00 0.00
OP8 0.00 0.00 N/A 0.00 N/A 0.00 0.00
TOTAL 5473.00 0.00 0.00 0.00 0.00

ACCOUNTING ROLLUP QUANTITY /SECOND /THREAD /COMMIT LATCH CNT /SECOND /SECOND /SECOND /SECOND
------------------------------------------------------- -------- ------- ------- ------- --------- -------- -------- -------- --------
ROLLUP THRESH REC WRITTEN 0.00 0.00 0.00 0.00 LC01-LC04 0.00 0.00 0.00 0.00
STORAGE THRESH REC WRITTEN 0.00 0.00 0.00 0.00 LC05-LC08 0.00 0.00 0.00 0.00
STALEN THRESH REC WRITTEN 26.00 0.00 0.14 0.00 LC09-LC12 0.00 0.00 0.00 0.00
RECS UNQUALIFIED FOR ROLLUP 0.00 0.00 0.00 0.00 LC13-LC16 0.00 0.00 0.00 0.00
LC17-LC20 0.00 0.00 0.01 0.00
LC21-LC24 0.00 0.00 0.00 0.00
LC25-LC28 0.00 0.00 0.00 0.00
LC29-LC32 0.00 0.00 0.00 0.00
LC254 0.00

LOCATION: OMPDBE3 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0)
GROUP: DBE3 STATISTICS REPORT - LONG
MEMBER: SE31 SUBSYSTEM: SE31

INTERVAL START : 10/14/13 21:54:00.00 SAMPLING START: 10/14/13 21:54:00.00 TOTAL THREADS : 186.00
INTERVAL END : 10/15/13 03:49:00.00 SAMPLING END : 10/15/13 03:49:00.00 TOTAL COMMITS : 25399.00
INTERVAL ELAPSED: 5:54:59.996886 OUTAGE ELAPSED: 0.000000 DATA SHARING MEMBER: N/A

MISCELLANEOUS VALUE
HIGH LOG RBA 00000000000E42EB3F7D
BYPASS COL 0.00
MAX SQL CASCADING LEVEL 0.00
MAX STOR LOB VALUES (MB) 0.00
MAX STOR XML VALUES (MB) 0.00
ARRAY EXPANSIONS 0.00
SPARSE IX DISABLED 0.00
SPARSE IX BUILT WF 0.00

LOCATION: OMPDBE3 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-12
GROUP: DBE3 STATISTICS REPORT - LONG REQUESTED FROM: NOT SPECIFIED
MEMBER: SE31 TO: NOT SPECIFIED
SUBSYSTEM: SE31 INTERVAL FROM: 10/14/13 21:54:00.00
DB2 VERSION: V11 SCOPE: MEMBER TO: 10/15/13 03:49:00.00

---- HIGHLIGHTS -----------------------------------------------------------------------------------------------
INTERVAL START : 10/14/13 21:54:00.00 SAMPLING START: 10/14/13 21:54:00.00 TOTAL THREADS : 186.00
INTERVAL END : 10/15/13 03:49:00.00 SAMPLING END : 10/15/13 03:49:00.00 TOTAL COMMITS : 25399.00
INTERVAL ELAPSED: 5:54:59.996886 OUTAGE ELAPSED: 0.000000 DATA SHARING MEMBER: N/A

DBM1 AND MVS STORAGE BELOW 2 GB QUANTITY DBM1 AND MVS STORAGE BELOW 2 GB CONTINUED QUANTITY
----------------------------------------------------------------------------------------------------------------
TOTAL DBM1 STORAGE BELOW 2 GB (MB) 6.52 24 BIT LOW PRIVATE (MB) 0.22
GETMAINED STORAGE (MB) 0.52 24 BIT HIGH PRIVATE( MB) 0.45
EDM POOL (MB) 0.00 24 BIT PRIVATE CURRENT HIGH ADDRESS 0000000003E000
TOTAL VARIABLE STORAGE (MB) 7.35 31 BIT EXTENDED LOW PRIVATE (MB) 74.39
TOTAL AGENT LOCAL STORAGE (MB) 0.32 31 BIT EXTENDED HIGH PRIVATE (MB) 27.69
TOTAL AGENT SYSTEM STORAGE (MB) 0.22 31 BIT PRIVATE CURRENT HIGH ADDRESS 0000000026F0D000
NUMBER OF PREFETCH ENGINES - 9.00 EXTENDED REGION SIZE (MAX) (MB) 1524.00
NUMBER OF DEFERRED WRITE ENGINES - 1.00 EXTENDED CSA SIZE (MB) 300.03
NUMBER OF CASTOUT ENGINES - 7.00
NUMBER OF GBP WRITE ENGINES - 1.00
NUMBER OF F-Lock/Notify Exit Engines - 2.00 MAX NUMBER OF POSSIBLE THREADS 3984
TOTAL NUMBER OF ACTIVE USER THREADS - 7.00 AVERAGE THREAD FOOTPRINT (MB) 0.25
NUMBER OF ALLIED THREADS - 7.00 AVERAGE THREAD FOOTPRINT (TYPE II) (MB) 0.09
NUMBER OF ACTIVE OATS - 0.00 MAX NUMBER OF POSSIBLE THREADS (TYPE II) 13581
NUMBER OF POOLED DBATS - 0.00
NUMBER OF PARALLEL CHILD THREADS - 0.00
RID POOL (MB) N/A
PIPE MANAGER SUB POOL (MB) N/A
LOCAL DYNAMIC STMTCNTL BLKS (MB) N/A
SYSTEM COPIES OF CACHED SQL STMTS (MB) N/A
IN USE STORAGE (MB) N/A
STATEMENTS COUNT N/A
HWM FOR ALLOCATED STATEMENTS (MB) N/A
STATEMENT COUNT AT HWM N/A
DATE AT HWM N/A
TIME AT HWM N/A
SYSTEM COPIES OF STATIC SQL (MB) N/A
IN USE STORAGE (MB) N/A
THREAD PLAN AND PACKAGE STORAGE (MB) 0.00
BUFFER MANAGER STORAGE CNTL BLKS (MB) 0.35
IN USE STORAGE (MB) N/A
TOTAL FIXED STORAGE (MB) 0.00
TOTAL GETMAINED STACK STORAGE (MB) 4.00
TOTAL STACK STORAGE IN USE (MB) 4.57
TOTAL AGENT NON-SYSTEM STORAGE (MB) 0.10
STORAGE MANAGER CONTROL BLOCKS (MB) 3.94
VIRTUAL BUFFER POOLS (MB) 66.22
VIRTUAL POOL CONTROL BLOCKS (MB) 1.11
CASTOUT BUFFERS (MB) 0.88
SHARED GETMAINED STORAGE (MB) 5.78
SHARED FIXED STORAGE (MB) 2.89
RID POOL (MB) 1.00
SHARED VARIABLE STORAGE (MB) 13.47
TOTAL AGENT LOCAL STORAGE (MB) 12.21

LOCATION: OMPDBE3 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-13
GROUP: DBE3 STATISTICS REPORT - LONG REQUESTED FROM: NOT SPECIFIED
MEMBER: SE31 TO: NOT SPECIFIED
SUBSYSTEM: SE31 INTERVAL FROM: 10/14/13 21:54:00.00
DB2 VERSION: V11 SCOPE: MEMBER TO: 10/15/13 03:49:00.00

---- HIGHLIGHTS -----------------------------------------------------------------------------------------------
INTERVAL START : 10/14/13 21:54:00.00 SAMPLING START: 10/14/13 21:54:00.00 TOTAL THREADS : 186.00
INTERVAL END : 10/15/13 03:49:00.00 SAMPLING END : 10/15/13 03:49:00.00 TOTAL COMMITS : 25399.00
INTERVAL ELAPSED: 5:54:59.996886 OUTAGE ELAPSED: 0.000000 DATA SHARING MEMBER: N/A

DBM1 STORAGE ABOVE 2 GB QUANTITY
----------------------------------------------------------------------------------------------------------------
GETMAINED STORAGE (MB) 408.62
FIXED STORAGE (MB) 6.64
VARIABLE STORAGE (MB) 30.20
COMPRESSION DICTIONARY (MB) 0.00
IN USE EDM DBD POOL (MB) 0.56
IN USE EDM STATEMENT POOL (MB) 0.72
IN USE EDM RDS POOL (MB) N/A
IN USE EDM SKELETON POOL (MB) 0.45
STORAGE MANAGER CONTROL BLOCKS (MB) 1.34
VIRTUAL BUFFER POOLS (MB) 66.22
VIRTUAL POOL CONTROL BLOCKS (MB) 1.11
CASTOUT BUFFERS (MB) 0.88
SHARED GETMAINED STORAGE (MB) 5.78
SHARED FIXED STORAGE (MB) 2.89
RID POOL (MB) 1.00
SHARED VARIABLE STORAGE (MB) 13.47
TOTAL AGENT LOCAL STORAGE (MB) 12.21

<table>
<thead>
<tr>
<th>Description</th>
<th>MB</th>
</tr>
</thead>
<tbody>
<tr>
<td>TOTAL AGENT SYSTEM STORAGE</td>
<td>5.57</td>
</tr>
<tr>
<td>TOTAL AGENT NON-SYSTEM STORAGE</td>
<td>6.64</td>
</tr>
<tr>
<td>THREAD COPIES OF CACHED SQL STMTS</td>
<td>N/A</td>
</tr>
<tr>
<td>IN USE STORAGE</td>
<td>0.00</td>
</tr>
<tr>
<td>STATEMENTS COUNT</td>
<td>0.00</td>
</tr>
<tr>
<td>HW FOR ALLOCATED STATEMENTS</td>
<td>0.01</td>
</tr>
<tr>
<td>STATEMENT COUNT AT HWM</td>
<td>0.00</td>
</tr>
<tr>
<td>DATE AT HWM</td>
<td>10/14/13</td>
</tr>
<tr>
<td>TIME AT HWM</td>
<td>21:56:56.96</td>
</tr>
<tr>
<td>DYNAMIC SMT CACHE CNTL BLKS</td>
<td>0.24</td>
</tr>
<tr>
<td>SYSTEM COPIES OF CACHED SQL STMTS</td>
<td>0.24</td>
</tr>
<tr>
<td>IN USE STORAGE</td>
<td>0.08</td>
</tr>
<tr>
<td>HW FOR ALLOCATED STATEMENTS</td>
<td>0.08</td>
</tr>
<tr>
<td>SYSTEM COPIES OF STATIC SQL</td>
<td>0.34</td>
</tr>
<tr>
<td>IN USE STORAGE</td>
<td>0.07</td>
</tr>
<tr>
<td>THREAD PLAN AND PACKAGE STORAGE</td>
<td>0.02</td>
</tr>
<tr>
<td>ARRAY VARIABLE STORAGE</td>
<td>0.00</td>
</tr>
<tr>
<td>SHARED STORAGE MANAGER CNTL BLKS</td>
<td>1.96</td>
</tr>
<tr>
<td>SHARED SYSTEM AGENT STACK STORAGE</td>
<td>256.00</td>
</tr>
<tr>
<td>STACK STORAGE IN USE</td>
<td>29.50</td>
</tr>
<tr>
<td>SHARED NON-SYSTEM AGENT STACK STORAGE</td>
<td>768.00</td>
</tr>
<tr>
<td>STACK STORAGE IN USE</td>
<td>3.50</td>
</tr>
<tr>
<td>TOTAL DIST STORAGE BELOW 2 GB</td>
<td>1.35</td>
</tr>
<tr>
<td>TOTAL GETMAINED STORAGE</td>
<td>0.00</td>
</tr>
<tr>
<td>TOTAL VARIABLE STORAGE</td>
<td>0.26</td>
</tr>
<tr>
<td>NUMBER OF ACTIVE CONNECTIONS</td>
<td>0.00</td>
</tr>
<tr>
<td>NUMBER OF INACTIVE CONNECTIONS</td>
<td>0.00</td>
</tr>
<tr>
<td>TOTAL FIXED STORAGE</td>
<td>0.08</td>
</tr>
<tr>
<td>TOTAL GETMAINED STACK STORAGE</td>
<td>1.00</td>
</tr>
<tr>
<td>TOTAL STACK STORAGE IN USE</td>
<td>0.86</td>
</tr>
<tr>
<td>SYSTEM AGENT STACK STORAGE IN USE</td>
<td>0.77</td>
</tr>
<tr>
<td>STORAGE CUSHION</td>
<td>329.63</td>
</tr>
<tr>
<td>24 BIT LOW PRIVATE</td>
<td>0.24</td>
</tr>
<tr>
<td>24 BIT HIGH PRIVATE</td>
<td>0.25</td>
</tr>
<tr>
<td>24 BIT PRIVATE CURRENT HIGH ADDRESS</td>
<td>0.00</td>
</tr>
<tr>
<td>31 BIT EXTENDED LOW PRIVATE</td>
<td>6.10</td>
</tr>
<tr>
<td>31 BIT EXTENDED HIGH PRIVATE</td>
<td>13.27</td>
</tr>
<tr>
<td>31 BIT PRIVATE CURRENT HIGH ADDRESS</td>
<td>0.00</td>
</tr>
<tr>
<td>EXTENDED REGION SIZE (MAX)</td>
<td>1524.00</td>
</tr>
<tr>
<td>REAL STORAGE IN USE</td>
<td>22.53</td>
</tr>
<tr>
<td>31 BIT IN USE</td>
<td>11.01</td>
</tr>
<tr>
<td>64 BIT IN USE</td>
<td>11.52</td>
</tr>
<tr>
<td>64 BIT THREAD AND SYSTEM ONLY</td>
<td>4.89</td>
</tr>
<tr>
<td>HW 64 BIT REAL STORAGE IN USE</td>
<td>112.87</td>
</tr>
<tr>
<td>AVERAGE THREAD FOOTPRINT</td>
<td>2.27</td>
</tr>
<tr>
<td>AUXILIARY STORAGE IN USE</td>
<td>206.49</td>
</tr>
<tr>
<td>31 BIT IN USE</td>
<td>64.12</td>
</tr>
<tr>
<td>64 BIT IN USE</td>
<td>142.37</td>
</tr>
<tr>
<td>HW 64 BIT AUX STORAGE IN USE</td>
<td>142.37</td>
</tr>
<tr>
<td>LOCATION: OMPDBE3</td>
<td></td>
</tr>
<tr>
<td>GROUP: BB3</td>
<td></td>
</tr>
<tr>
<td>SUBSYSTEM: SE31</td>
<td></td>
</tr>
<tr>
<td>DB2 VERSION: V11</td>
<td></td>
</tr>
<tr>
<td>INTERVAL END: 10/15/13 03:49:00.00</td>
<td></td>
</tr>
<tr>
<td>TOTAL COMITS: 25399.00</td>
<td></td>
</tr>
<tr>
<td>INTERVAL ELAPSED: 5:54:59.996886</td>
<td></td>
</tr>
<tr>
<td>OUTAGE ELAPSED: 0.00</td>
<td></td>
</tr>
<tr>
<td>DATA SHARING MEMBER: N/A</td>
<td></td>
</tr>
<tr>
<td>COMMON STORAGE BELOW AND ABOVE 2 GB</td>
<td></td>
</tr>
<tr>
<td>EXTENDED CSA SIZE</td>
<td>300.03</td>
</tr>
<tr>
<td>FIXED POOL BELOW</td>
<td>0.75</td>
</tr>
<tr>
<td>VARIABLE POOL BELOW</td>
<td>0.68</td>
</tr>
<tr>
<td>GETMAINED BELOW</td>
<td>0.10</td>
</tr>
<tr>
<td>FIXED POOL ABOVE</td>
<td>2.27</td>
</tr>
<tr>
<td>VARIABLE POOL ABOVE</td>
<td>70.03</td>
</tr>
<tr>
<td>SUBSYSTEM SHARED STORAGE ABOVE 2 GB</td>
<td></td>
</tr>
<tr>
<td>REAL STORAGE IN USE</td>
<td>2.88</td>
</tr>
<tr>
<td>FIXED THREAD AND SYSTEM</td>
<td>2.28</td>
</tr>
<tr>
<td>SHARE STACK STORAGE</td>
<td>0.59</td>
</tr>
<tr>
<td>AVERAGE THREAD FOOTPRINT</td>
<td>0.41</td>
</tr>
<tr>
<td>AUXILIARY STORAGE IN USE</td>
<td>27.87</td>
</tr>
<tr>
<td>SHARE STACK STORAGE</td>
<td>14.57</td>
</tr>
<tr>
<td>SHARE STACK STORAGE</td>
<td>13.30</td>
</tr>
</tbody>
</table>
### Storage Manager Control Blocks Above (MB)
- 0.19

### Real Log Manager Write Buffers Above (MB)
- 4.00

### Real Log Manager Control Blocks Above (MB)
- 0.00

### Aux Log Manager Control Blocks Above (MB)
- 0.00

### Real Storage in Use (MB)
- 2.35

### Average Thread Footprint (MB)
- 0.34

### Auxiliary Storage in Use (MB)
- 3.50

### MVS LPAR Shared Storage Above 2 GB Quantity

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Storage Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.00</td>
<td>Shared Memory Objects</td>
</tr>
</tbody>
</table>

### 31/64-Bit Private (DBM1)
- 22.53

### 64 Bit Shared Storage
- 1540096.00

### HWM for 64 Bit Shared Storage
- 1540096.00

### 64 Bit Shared Storage Backed in Real
- 18.76

### 64 Bit Shared Storage Paged in From Aux
- 35.46

### 64 Bit Shared Storage Paged Out to Aux
- 194.08

### Location: OMPDBE3
- GROUP: OMEGAMON XE for DB2 Performance Expert (V5R2M0)
- MEMBER: SE31
- SUBSYSTEM: SE31

### Interval Start: 10/14/13 21:54:00.00
### Interval End: 10/15/13 03:49:00.00
### Total Threads: 186.00
### Total Commits: 25399.00
### Interval Elapsed: 5:54:59.996886
### Outage Elapsed: 0.000000

### Workfile Database Quantity /Second /Thread /Commit CPU and Storage Metrics Quantity

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Storage Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>134.3M</td>
<td>Total Storage Config (KB)</td>
</tr>
<tr>
<td>N/A</td>
<td>TOT DGTT Stor Config (KB)</td>
</tr>
<tr>
<td>N/A</td>
<td>TOT WF Stor Config (KB)</td>
</tr>
<tr>
<td>90.00</td>
<td>Total Storage Threshold (%)</td>
</tr>
<tr>
<td>N/A</td>
<td>MAX TOTAL Storage Used (KB)</td>
</tr>
<tr>
<td>N/A</td>
<td>MAX DGTT Stor Used (KB)</td>
</tr>
<tr>
<td>N/A</td>
<td>CUR TOTAL Stor Used (KB)</td>
</tr>
<tr>
<td>N/A</td>
<td>CUR DGTT Stor Used (KB)</td>
</tr>
<tr>
<td>N/A</td>
<td>CUR WF Stor Used (KB)</td>
</tr>
<tr>
<td>N/A</td>
<td>STORAGE IN 4K TS (KB)</td>
</tr>
<tr>
<td>N/A</td>
<td>STORAGE IN 32K TS (KB)</td>
</tr>
<tr>
<td>N/A</td>
<td>4K Used Instead of 32K TS</td>
</tr>
<tr>
<td>N/A</td>
<td>32K Used Instead of 4K TS</td>
</tr>
<tr>
<td>0.00</td>
<td>MAX ACTIVE (CM) IN-MEM</td>
</tr>
<tr>
<td>0.00</td>
<td>MAX ACTIVE (CM) IN-SORT</td>
</tr>
<tr>
<td>0.00</td>
<td>CUR ACTIVE (CM) IN-MEM</td>
</tr>
<tr>
<td>0.00</td>
<td>CUR ACTIVE (CM) IN-SORT</td>
</tr>
<tr>
<td>0.00</td>
<td>MAX STOR (CM) IN-MEM</td>
</tr>
<tr>
<td>0.00</td>
<td>MAX STOR (CM) IN-SORT</td>
</tr>
<tr>
<td>0.00</td>
<td>CUR STOR (CM) IN-MEM</td>
</tr>
<tr>
<td>0.00</td>
<td>CUR STOR (CM) IN-SORT</td>
</tr>
<tr>
<td>0.00</td>
<td>MAX STOR (SORT) IN-MEM</td>
</tr>
<tr>
<td>0.00</td>
<td>MAX STOR (SORT) IN-SORT</td>
</tr>
<tr>
<td>0.00</td>
<td>CUR STOR (SORT) IN-MEM</td>
</tr>
<tr>
<td>0.00</td>
<td>CUR STOR (SORT) IN-SORT</td>
</tr>
</tbody>
</table>

### Location: OMPDBE3
- GROUP: OMEGAMON XE for DB2 Performance Expert (V5R2M0)
- MEMBER: SE31
- SUBSYSTEM: SE31

### Interval Start: 10/14/13 21:54:00.00
### Interval End: 10/15/13 03:49:00.00
### Total Commits: 25399.00

### Interval Elapsed: 5:54:59.996886
### Outage Elapsed: 0.000000

### Data Sharing Member: N/A

### Short-on-Storage Metrics

#### GETPAGE Request

- **Random**: 232.00
- **Sequential**: 464.00

#### Page-Ins Required for Write

- **Random**: 0.00
- **Sequential**: 0.00

#### Page-Ins Required for Read

- **Random**: 1.00

#### Critical Shortages

- **Random**: 0.00
- **Sequential**: 0.00

#### Abends Due to Shortages

- **Random**: 0.00
- **Sequential**: 0.00

---

### Buffers Allocated - VPOOL

- **Total**: 1000.00
- **Used**: 3648.00

---

### Agent Storage Config (KB)

- **Used**: N/A
- **Unused**: N/A
- **Total**: N/A

---

### Agent Storage Threshold (%)

- **Allocated**: N/A
- **Free**: N/A

---

### Short-On-Storage Metrics

<table>
<thead>
<tr>
<th>Metric</th>
<th>Quantity</th>
<th>/SECOND</th>
<th>/THREAD</th>
<th>/COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Current Active Buffers</td>
<td>55.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Number of Dataset Opens</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Buffers Allocated - VPOOL</td>
<td>5000.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>DFSM Migrated Dataset</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>VPOOL Expans. or Contract.</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>VPOOL Expans. Failures</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Concur. Pref. I/O Streams-HWM</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Pref. I/O Streams Reduction</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>平行 Query Requests</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Parall. Query Req. Reduction</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Pref. Quant. Reduced To 1/2</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Pref. Quant. Reduced To 1/4</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Number of LPL Inserts</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Min Buffers on SLRU</td>
<td>783.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Max Buffers on SLRU</td>
<td>784.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>SLRU Length Equals VSSEQT</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>GetPage Request Random on SLRU</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>

---

### Page Operations

<table>
<thead>
<tr>
<th>Metric</th>
<th>Quantity</th>
<th>/SECOND</th>
<th>/THREAD</th>
<th>/COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Buffer Updates</td>
<td>388.00</td>
<td>0.02</td>
<td>2.97</td>
<td>0.02</td>
</tr>
<tr>
<td>Pages Written</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Synchronous Writes</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Pages Written Per Write I/O</td>
<td>430.00</td>
<td>0.02</td>
<td>2.31</td>
<td>0.02</td>
</tr>
<tr>
<td>Number of Castout I/O</td>
<td>293.00</td>
<td>0.01</td>
<td>1.58</td>
<td>0.01</td>
</tr>
<tr>
<td>Page-Ins Required for Write</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>

---

### Location

**OMEGAMON XE for DB2 Performance Expert (V5R2M0)**

**GROUP**: DBE3

**Member**: SE31

**Subsystem**: SES1

**DB Version**: V11

---

**Highlights**

**Interval Start**: 10/14/13 21:54:00.00

**Sampling Start**: 10/14/13 21:54:00.00

**Total Commits**: 25399.00

---

**BP0 General**

- **Current Active Buffers**: 55.00
- **Unavailable Buffer-VPOOL Full**: 0.00
- **Number of Dataset Opens**: 0.00
- **Buffers Allocated - VPOOL**: 5000.00

---

**BP0 Read Operations**

- **BP0 GETPAGE Request**
  - **Random**: 40417.00
  - **Sequential**: 3993.00
- **BP0 Buffer Hit Ratio (%)**
  - **Random**: 100.00
  - **Sequential**: 98.85

---

**BP32K General**

- **Current Active Buffers**: 4.00
- **Unavailable Buffer-VPOOL Full**: 0.00
- **Number of Dataset Opens**: 0.00
- **Buffers Allocated - VPOOL**: 1000.00

---

**BP32K Read Operations**

- **BP32K GETPAGE Request**
  - **Random**: 464.00
  - **Sequential**: 232.00
- **BP32K Buffer Hit Ratio (%)**
  - **Random**: 100.00
  - **Sequential**: 98.96

---

**Subsystem**: SE31

**Interval From**: 10/14/13 21:54:00.00

**Sampling Start**: 10/14/13 21:54:00.00

**Total threads**: 186.00

---

**Interval Elapsed**: 5:54:59.996886

**Outage Elapsed**: 0.000000

---

**Data Sharing Member**: N/A

---

**Location**: OMPDBE3

**OMEGAMON XE for DB2 Performance Expert (V5R2M0)**

**GROUP**: DBE3

**Member**: SE31

**Subsystem**: SES1

**DB Version**: V11
<table>
<thead>
<tr>
<th>Metric</th>
<th>Value 1</th>
<th>Value 2</th>
<th>Value 3</th>
<th>Value 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>DFHSM RECALL TIMEOUTS</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DFHSM MIGRATED DATASET</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>VPOOL EXPANS. OR CONTRACT.</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>VPOOL EXPANS. FAILURES</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>CONC/PREF.1/O STREAMS-HM</td>
<td>0.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>PREF.1/O STREAMS REDUCTION</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>PARALLEL QUERY REQUESTS</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>PARAL.QUERY_REQ.REDUCTION</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>PREF.QUANT.REDUCED TO 1/4</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>NUMBER OF CASTOUT I/O</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>MIN BUFFERS ON SLRU</td>
<td>24.72</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>MAX BUFFERS ON SLRU</td>
<td>25.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>SLRU LENGTH EQUALS VPSQ2T</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>GETPAGE REQ RANDOM ON SLRU</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>NUMBER OF DATASET OPENS</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>BPOOL HIT RATIO (%)</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>DB2 VERSION: V11</td>
<td>SUBSYSTEM: SE31</td>
<td>INTERVAL FROM: 10/14/13 21:54:00.00</td>
<td>REQUESTED FROM: NOT SPECIFIED</td>
<td>TO: NOT SPECIFIED</td>
</tr>
<tr>
<td>LOCATION: OMPB3</td>
<td>OMAGMEN XE FOR DBZ PERFORMANCE EXPERT (VS02MD)</td>
<td>MEMBER: SE31</td>
<td>INTERVAL FROM: 10/14/13 21:54:00.00</td>
<td>TO: 10/15/13 03:49:00.00</td>
</tr>
<tr>
<td>BUFFER UPDATES</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>PAGES WRITTEN</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>BUF.UPDATE/PAGES WRITTEN</td>
<td>N/C</td>
<td>N/C</td>
<td>N/C</td>
<td>N/C</td>
</tr>
<tr>
<td>SYNCHRONOUS WRITES</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>SYNCHRONOUS WRITES</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>SYNCHRONOUS WRITES</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>ASYNNCHRONOUS WRITES</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>PAGE WRITTEN PER WRITE 1/0</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>PAGE WRITEN FOR CASTOUT 1/0</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DM THRESHOLD</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>PAGE-INS REQUIRED FOR WRITE</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>

REG.PAGE LIST (RPL) REQUEST
NUMBER OF PAGES RETR.FROM GBP
PAGES CASTOUT
UNLOCK CASTOUT
READ CASTOUT CLASS
READ DIRECTORY INFO
READ STORAGE STATISTICS
REGISTER PAGE
DELETE NAME
ASYNC GBP REQUESTS
EXPLICIT X-INVALIDATIONS
CASTOUT CLASS THRESHOLD
GROUP BP CASTOUT THRESHOLD
GBP CHECKPOINTS TRIGGERED
WRITE TO SEC-GBP FAILED
CPL CHECKS SUSPENDED
DELETE NAME LIST SEC-GBP
DELETE NAME FROM SEC-GBP
UNLOCK CASTOUT STATS SEC-GBP
ASYNCH SEC-GBP REQUESTS
LOCATION: OMPDBE3
GROUP: DBE3
MEMBER: SE31
SUBSYSTEM: SE31
INTERVAL START : 10/14/13 21:54:00.00
INTERVAL END : 10/15/13 03:49:00.00
INTERVAL ELAPSED: 5:54:59.996886
OUTAGE ELAPSED: 0.000000
DATA SHARING MEMBER: N/A

GROUP TOTAL QUANTITY /SECOND /THREAD /COMMIT
GROUP TOTAL CONTINUED QUANTITY /SECOND /THREAD /COMMIT
GROUP BP R/W RATIO (%) 0.00 N/A N/A N/A
GBP SYN.READ[XI] HIT RATIO(%) 0.00 N/A N/A N/A
GBP-DEPENDENT GETPAGES 32970.00 1.55 177.26 1.30
SYN.READ[XI]-DATA RETURNED 0.00 0.00 0.00 0.00
SYN.READ[XI]-NO DATA RETURN 313.00 0.01 1.68 0.01
SYN.READ[NF]-DATA RETURNED 0.00 0.00 0.00 0.00
SYN.READ[NF]-NO DATA RETURN 0.00 0.00 0.00 0.00
UNREGISTER PAGE 0.00 0.00 0.00 0.00

CLEAN PAGES SYNCR.WRITTEN 0.00 0.00 0.00 0.00
CLEAN PAGES ASYNC.WRITN 0.00 0.00 0.00 0.00
REG.PAGE LIST (RPL) REQUEST 0.00 0.00 0.00 0.00
NUMBER OF PAGES RETR.FROM GBP 0.00 0.00 0.00 0.00
PAGES CASTOUT 430.00 0.02 2.31 0.02
UNLOCK CASTOUT 293.00 0.01 1.58 0.01
READ CASTOUT CLASS 233.00 0.01 1.25 0.01
READ DIRECTORY INFO 0.00 0.00 0.00 0.00
READ STORAGE STATISTICS 1668.00 0.08 8.97 0.07
REGISTER PAGE 0.00 0.00 0.01 0.00
DELETE NAME 0.00 0.00 0.00 0.00
ASYNC GBP REQUESTS 2319.00 0.11 12.47 0.09
EXPLICIT X-INVALIDATIONS 0.00 0.00 0.00 0.00
CASTOUT CLASS THRESHOLD 0.00 0.00 0.00 0.00
GROUP BP CASTOUT THRESHOLD 0.00 0.00 0.00 0.00
GBP CHECKPOINTS TRIGGERED 0.00 0.00 0.00 0.00
WRITE TO SEC-GBP FAILED 0.00 0.00 0.00 0.00
CPL CHECKS SUSPENDED 0.00 0.00 0.00 0.00
DELETE NAME LIST SEC-GBP 0.00 0.00 0.00 0.00
UNLOCK CASTOUT STATS SEC-GBP 0.00 0.00 0.00 0.00
ASYNC SEC-GBP REQUESTS 0.00 0.00 0.00 0.00

LOCATION: OMPDBE3
GROUP: DBE3
MEMBER: SE31
SUBSYSTEM: SE31
INTERVAL FROM: 10/14/13 21:54:00.00
DB2 VERSION: V11
SCOPE: MEMBER
TO: 10/15/13 03:49:00.00

--- HIGHLIGHTS ---

INTERVAL START : 10/14/13 21:54:00.00
INTERVAL END : 10/15/13 03:49:00.00
TOTAL THREADS : 186.00
TOTAL COMMITS : 25399.00
INTERVAL ELAPSED: 5:54:59.996886
OUTAGE ELAPSED: 0.000000

48-14 OMEGAMON XE for DB2 PE & PM: Report Reference
Chapter 49. Statistics Report and Trace Blocks

This section shows the individual blocks presented by OMEGAMON XE for DB2 PE reports and traces together with a short explanation of each field. The examples shown are taken from the statistics long reports and traces.

The layout of the Statistics report and trace blocks is the same. Statistics traces show times and events as delta records describing the activity between two consecutive DB2 record pairs. Statistics reports show times and events over a user-specified period of time.

Each block is presented in the default layout. Some block can have columns, rows, or fields that are not included in the default layout. You can include columns, rows, and fields not shown in the default layouts with user-tailored reporting (UTR).

In the short report and trace, field names can differ slightly from the names shown in the long report or trace.

If a counter value or specific information in reports, in windows, or on panels is not shown, the following notation is used to indicate the reason:

N/A Not applicable is shown if DB2 never produces a counter value in a specific context. Examples are:
- A counter is not available in one DB2 version.
- Counters are mutually exclusive.

N/C Not calculated is shown for a derived field where the value cannot be calculated or is useless. Examples are:
- A divide by zero (percentages, ratios).
- Suppression of negative elapsed time values.
- Required counter values for calculation marked as N/A or N/P.
- Insufficient data or small counter values to allow significant statements (meaningless or misleading averages).

N/P Not present is shown for a field where DB2 can present values, but does not in this instance. Examples are:
- When counter values are not generated because of operational conditions (a trace class is not active).
- An application does not provide a value because it is optional.
"Buffer Pool Read" on page 49-20
This topic shows detailed information about “Statistics - Buffer Pool Read”.

"Buffer Pool Sort/Merge" on page 49-28
This topic shows detailed information about “Statistics - Buffer Pool Sort/Merge”.

"Buffer Pool Write" on page 49-31
This topic shows detailed information about “Statistics - Buffer Pool Write”.

"Common Storage Below and Above 2 GB (DB2 10 or later)” on page 49-37
This topic shows detailed information about “Statistics - Common Storage Below and Above 2 GB (DB2 10 or later)”.

"CPU and Storage Metrics” on page 49-39
This topic shows detailed information about “Statistics - CPU and Storage Metrics”.

"CPU Times” on page 49-41
This topic shows detailed information about “Statistics - CPU Times”.

"Data Capture” on page 49-44
This topic shows detailed information about “Statistics - Data Capture”.

"Data Set Statistics” on page 49-45
This topic shows detailed information about “Statistics - Data Set Statistics”.

"Data Sharing Locking” on page 49-47
This topic shows detailed information about “Statistics - Data Sharing Locking”.

"DBM1 and MVS Storage Below 2 GB” on page 49-52
This topic shows detailed information about “Statistics - DBM1 and MVS Storage Below 2 GB”.

"DBM1 Storage Above 2 GB” on page 49-59
This topic shows detailed information about “Statistics - DBM1 Storage Above 2 GB”.

"Dynamic SQL Statement” on page 49-65
This topic shows detailed information about “Statistics - Dynamic SQL Statement”.

"DB2 API” on page 49-68
This topic shows detailed information about “Statistics - DB2 API”.

"DB2 Commands” on page 49-69
This topic shows detailed information about “Statistics - DB2 Commands”.

"DIST Storage Above 2 GB (DB2 10 or later)” on page 49-76
This topic shows detailed information about “Statistics - DIST Storage Above 2 GB (DB2 10 or later)”.

"DIST and MVS Storage Below 2 GB (DB2 10 or later)” on page 49-77
This topic shows detailed information about “Statistics - DIST and MVS Storage Below 2 GB (DB2 10 or later)”.

"DRDA Remote Locations” on page 49-80
This topic shows detailed information about “Statistics - DRDA Remote Locations”.

"EDM Pool Activity” on page 49-86
"Global DDF Activity” on page 49-98
This topic shows detailed information about “Statistics - Global DDF Activity”.

"Group Buffer Pool Activity” on page 49-103
This topic shows detailed information about “Statistics - Group Buffer Pool Activity”.

"Highlights” on page 49-115
This topic shows detailed information about “Statistics - Highlights”.

49-2 OMEGAMON XE for DB2 PE & PM: Report Reference
This topic shows detailed information about “Statistics - Accelerator”.

The Statistics Accelerator report block is shown for each accelerator that provided services to the DB2 subsystem within the reported interval. The accelerator name is shown in the header line of each column together with the labels ACCELERATION and CONTINUED.

Note:
- The DB2 subsystem is connected with an accelerator via DRDA when submitting SQL queries. The performance counters of the DRDA connection (rows from CONNECTS TO ACCELERATOR to ROWS RECEIVED FROM ACCELERATOR) are collected during the termination of the associated thread. The other statistics counters are collected periodically from the accelerator.
- The full accelerator name is shown in the header if it does not exceed 16 characters. Otherwise, the name is replaced by a short name in the header and the long name is displayed at the end of the report.

### Statistics - Accelerator

The field labels shown in the following sample layout of “Statistics - Accelerator” are described in the following section.

<table>
<thead>
<tr>
<th>SIM03 ACCELERATION QUANTITY</th>
<th>SIM03 CONTINUED QUANTITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>QUERIES SUCCESSFULLY EXECUTED</td>
<td>266.00</td>
</tr>
<tr>
<td>QUERIES FAILED TO EXECUTE</td>
<td>1.00</td>
</tr>
<tr>
<td>ACCELERATOR IN INVALID STATE</td>
<td>0.00</td>
</tr>
<tr>
<td>CURRENTLY EXECUTING QUERIES</td>
<td>0.02</td>
</tr>
<tr>
<td>MAXIMUM EXECUTING QUERIES</td>
<td>2.00</td>
</tr>
<tr>
<td>CONNECTS TO ACCELERATOR</td>
<td>1.00</td>
</tr>
<tr>
<td>REQUESTS SENT TO ACCELERATOR</td>
<td>2.00</td>
</tr>
<tr>
<td>TIMED OUT</td>
<td>0.00</td>
</tr>
<tr>
<td>FAILED</td>
<td>0.00</td>
</tr>
<tr>
<td>BYTES SENT TO ACCELERATOR</td>
<td>2385.00</td>
</tr>
<tr>
<td>BYTES RECEIVED FROM ACCELERATOR</td>
<td>976.00</td>
</tr>
<tr>
<td>MESSAGES SENT TO ACCELERATOR</td>
<td>11.00</td>
</tr>
<tr>
<td>MESSAGES RECEIVED FROM ACCELERATOR</td>
<td>11.00</td>
</tr>
<tr>
<td>BLOCKS SENT TO ACCELERATOR</td>
<td>0.00</td>
</tr>
<tr>
<td>BLOCKS RECEIVED FROM ACCELERATOR</td>
<td>0.00</td>
</tr>
<tr>
<td>ROWS SENT TO ACCELERATOR</td>
<td>0.00</td>
</tr>
<tr>
<td>ROWS RECEIVED FROM ACCELERATOR</td>
<td>0.00</td>
</tr>
<tr>
<td>TCP/IP SERVICES ELAPSED TIME</td>
<td>5:15.06486</td>
</tr>
<tr>
<td>WAIT TIME IN ACCELERATOR</td>
<td>0.000800</td>
</tr>
</tbody>
</table>

### QUERIES SUCCESSFULLY EXECUTED

The number of queries (sent by this DB2 system since accelerator start) that were successfully executed in the accelerator.

**Field Name:** Q8STSREQ

### QUERIES FAILED TO EXECUTE

The number of queries (sent by this DB2 system since accelerator start) that failed to be successfully executed for any reason, including the accelerator being in an invalid state.

**Field Name:** Q8STFREQ

### QUERIES FAILED TO EXECUTE - ACCELERATOR IN INVALID STATE

The number of queries (sent by this DB2 system since accelerator start) that failed to be successfully executed, for example, because the accelerator was in an invalid state.
Accelerator

Field Name: Q8STFINV
CURRENTLY EXECUTING QUERIES
The number of currently executing queries in the accelerator. This includes
the queries from all the DB2 systems connected to this accelerator.

Field Name: Q8STACTV
MAXIMUM EXECUTING QUERIES
The maximum number of queries executing in the accelerator at any time
since accelerator start. This includes the queries from all the DB2 systems
connected to this accelerator.

Field Name: Q8STMAXA
CONNECTS TO ACCELERATOR
The number of connects to the accelerator from this DB2 system.

Field Name: Q8STCONN
REQUESTS SENT TO ACCELERATOR
The number of Distributed Relational Database Architecture (DRDA)
requests sent by this DB2 system to the accelerator.

Field Name: Q8STREQ
REQUESTS SENT TO ACCELERATOR - TIMED OUT
The number of connections that were timed out when this DB2 system sent
requests to the accelerator.

Field Name: Q8STTOUT
REQUESTS SENT TO ACCELERATOR - FAILED
The number of connections that failed when this DB2 system sent requests
to the accelerator.

Field Name: Q8STFAIL
BYTES SENT TO ACCELERATOR
The total number of bytes sent to the accelerator.

Field Name: Q8STBYTS
BYTES RECEIVED FROM ACCELERATOR
The total number of bytes received from the accelerator.

Field Name: Q8STBYTR
MESSAGES SENT TO ACCELERATOR
The total number of messages sent to the accelerator.

Field Name: Q8STMSGS
MESSAGES RECEIVED FROM ACCELERATOR
The total number of messages received from the accelerator.

Field Name: Q8STMSGR
BLOCKS SENT TO ACCELERATOR
The total number of blocks sent to the accelerator.
Field Name: Q8STBLKS

**BLOCKS RECEIVED FROM ACCELERATOR**

The total number of blocks received from the accelerator.

Field Name: Q8STBLKR

**ROWS SENT TO ACCELERATOR**

The total number of rows sent to the accelerator.

Field Name: Q8STROWS

**ROWS RECEIVED FROM ACCELERATOR**

The total number of rows received from the accelerator.

Field Name: Q8STROWR

**TCP/IP SERVICES ELAPSED TIME**

The accumulated accelerator services TCP/IP elapsed time measured in DB2. It starts when sending the requests to the accelerator and ends when receiving the results from the accelerator.

Field Name: Q8STTELA

**WAIT TIME IN ACCELERATOR**

The accumulated wait time spent in the accelerator when executing requests from the DB2 subsystem.

Field Name: Q8STAWAT

**AVG. QUEUE LENGTH (LAST 3 HOURS)**

The average query queue length during the last 3 hours at the accelerator.

Field Name: Q8STAVGQ03

**AVG. QUEUE LENGTH (LAST 24 HOURS)**

The average query queue length during the last 24 hours at the accelerator.

Field Name: Q8STAVGQ24

**MAXIMUM QUEUE LENGTH**

The high watermark of the query queue length at the accelerator.

Field Name: Q8STMAXQ

**AVG QUEUE WAIT ELAPSED TIME**

The average wait time at the accelerator query queue.

Field Name: Q8STQUEW

**MAX QUEUE WAIT ELAPSED TIME**

The maximum wait time at the accelerator query queue.

Field Name: Q8STQUEM

**WORKER NODES**

The number of active worker nodes.

Field Name: Q8STWNOD

**WORKER NODES AVG CPU UTILIZATION (%)**
Accelerator

The average CPU utilization on accelerator worker nodes.
**Field Name:** Q8STWCPU

**COORDINATOR AVG CPU UTILIZATION (%)**
The average CPU utilization on the accelerator coordinator node.
**Field Name:** Q8STCCPU

**DISK STORAGE AVAILABLE (MB)**
The disk storage (MB) available at the accelerator.
**Field Name:** Q8STDSKA

**DISK STORAGE AVAILABLE - IN USE (%)**
The disk storage in-use at the accelerator, expressed as a percentage (%).
**Field Name:** Q8STDSKU

**DISK STORAGE AVAILABLE - IN USE FOR DATABASE (MB)**
The disk storage in-use for the DB2 subsystem.
**Field Name:** Q8STDSKB

**DATA SLICES**
The number of data slices at the accelerator.
**Field Name:** Q8STNMDS

**DATA SKEW**
When table data is loaded into the accelerator, it may be unevenly distributed across the different data slices on the disks. This disparity is called data skew. The counter represents the accumulated skew over all tables that belong to the DB2 subsystem. The skew of a table is the ratio that shows how uneven the data slices are, as calculated by \((\text{maximum data slice size} - \text{minimum data slice size}) / \text{median data slice size}\).

A high value indicates that data reorganization can improve disk utilization and query performance.
**Field Name:** Q8STSKEW

**PROCESSORS**
The total amount of processors within the accelerator.
**Field Name:** Q8STCORS

**ELAPSED TIME IN ACCELERATOR**
The accumulated elapsed time spent in the accelerator when executing requests from the DB2 subsystem.
**Field Name:** Q8STAELA

**CPU TIME SPENT IN ACCELERATOR**
The accumulated CPU time spent in the accelerator when executing requests from the DB2 subsystem.
**Field Name:** Q8STACPU
Accounting Rollup

This topic shows detailed information about “Statistics - Accounting Rollup”.

Statistics - Accounting Rollup

The field labels shown in the following sample layout of “Statistics - Accounting Rollup” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Accountant Rollup</th>
<th>Quantity /Second</th>
<th>Thread</th>
<th>Commit</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROLLUP THRESH RECS WRITTEN</td>
<td>4.00</td>
<td>0.02</td>
<td>N/C</td>
<td>0.12</td>
</tr>
<tr>
<td>STORAGE THRESH RECS WRITTEN</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>STALEN THRESH RECS WRITTEN</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>RECS UNQUALIFIED FOR ROLLUP</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
</tbody>
</table>

**ROLLUP THRESH RECS WRITTEN**

The number of roll-up accounting records written due to roll-up threshold exceeded.

Field Name: QWSDARCTH

**STORAGE THRESH RECS WRITTEN**

The number of roll-up accounting records written due to roll-up accounting storage threshold exceeded.

Field Name: QWSDARSG

**STALEN THRESH RECS WRITTEN**

The number of roll-up accounting records written due to staleness threshold exceeded.

Field Name: QWSDARSST

**RECS UNQUALIFIED FOR ROLLUP**

The number of records that failed to qualify for accounting roll-up because all roll-up key fields are equal to NULL or because of NULL values that are not permitted.

Field Name: QWSDARIR
Aggregated Accounting Statistics

This topic shows detailed information about “Statistics - Aggregated Accounting Statistics”.

IFCID 369 contains aggregated Accounting data from IFCID 3 listed by connection type.

IFCID 369 is started using the START TRACE command for STATISTICS CLASS(9). IFCID 3 must also be enabled (ACCOUNTING CLASS(1)) to get 369 trace records. IFCID 369 values are aggregated each time an IFCID 3 is written and contain total values. In the Statistics reports, IFCID 369 delta values are calculated to show which IFCID 3 events occurred in a Statistics interval.

In contrast to Accounting reports, the IFCID 369 statistics performance metrics cannot distinguish between IFCID 3 records for parallel and non-parallel threads. In Accounting, the elapsed times of parallel threads are derived from IFCID 3 of the originating thread. In IFCID 369, all times (even the elapsed times of the originating thread and parallel subtasks) are aggregated and shown as such in the Statistics report. That is why IFCID 369 can provide more diagnostics on what is currently happening in a DB2 subsystem and whether there are bottlenecks (also for parallel subtasks).

For more information on the Accounting fields referred to in the field descriptions below, see:

- “Times - Class 1 - Application Time” on page 5-195
- “Times - Class 2 - DB2 Time” on page 5-202

Statistics - Aggregated Accounting Statistics

The field labels shown in the following sample layout of “Statistics - Aggregated Accounting Statistics” are described in the following section.

<table>
<thead>
<tr>
<th>CONNTYPE</th>
<th>CL1 ELAPSED</th>
<th>CL1 CPU</th>
<th>CL1 SE CPU</th>
<th>CL2 ELAPSED</th>
<th>CL2 CPU</th>
<th>CL2 SE CPU</th>
<th>CL3 SUSP</th>
<th>CL2 NOT ACC</th>
<th>QUANTITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>BATCH</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
</tr>
<tr>
<td>CICS</td>
<td>N/P</td>
<td>N/P</td>
<td>N/P</td>
<td>N/P</td>
<td>N/P</td>
<td>N/P</td>
<td>N/P</td>
<td>N/P</td>
<td>N/P</td>
</tr>
<tr>
<td>DDF</td>
<td>3:20.927303</td>
<td>8.240478</td>
<td>55.917303</td>
<td>8.697064</td>
<td>6.74272</td>
<td>7.368151</td>
<td>7.368151</td>
<td>15895.00</td>
<td></td>
</tr>
<tr>
<td>IMS</td>
<td>N/P</td>
<td>N/P</td>
<td>N/P</td>
<td>N/P</td>
<td>N/P</td>
<td>N/P</td>
<td>N/P</td>
<td>N/P</td>
<td>N/P</td>
</tr>
<tr>
<td>RRSASF</td>
<td>1:04.091521</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
</tr>
<tr>
<td>UTILITY</td>
<td>21.317271</td>
<td>0.431343</td>
<td>21.317271</td>
<td>1.224435</td>
<td>0.019245</td>
<td>0.019245</td>
<td>0.987360</td>
<td>0.047829</td>
<td>72.00</td>
</tr>
</tbody>
</table>

**CL1 ELAPSED**

The class 1 elapsed time aggregated by connection type. See also the description of Accounting field ADRECETT.

**Note:** In contrast to ADRECETT, elapsed times of parallel records are included in SDRECETT.

**Field Name:** SDRECETT

**CL1 CPU**

The class 1 CPU time aggregated by connection type. See also the description of Accounting field ADCPUT.

**Field Name:** SDCPUT

**CL1 SE CPU**
Aggregated Accounting Statistics

The sum of several accumulated CPU times consumed while running on an IBM specialty engine in all environments and aggregated by connection type. See also the description of Accounting field AWACC1Z.

Field Name: SWACC1Z

CL2 ELAPSED

The class 2 elapsed time aggregated by connection type. See also the description of Accounting field ADDB2ETT.

Note: In contrast to ADDB2ETT, elapsed times of parallel records are included in SDDB2ETT.

Field Name: SDDB2ETT

CL2 CPU

The class 1 CPU time aggregated by connection type. See also the description of Accounting field ADDBCPUT.

Field Name: SDDBCPUT

CL2 SE CPU

The accumulated and consumed class 2 time on an IBM specialty engine aggregated by connection type. See also the description of Accounting field AWACC2Z.

Field Name: SWACC2Z

CL3 SUSP

The waiting time for all types of class 3 suspensions aggregated by connection type. See also the description of Accounting field ADTSUST.

Field Name: SDTSUST

CL2 NOT ACC

The time not accounted in DB2 and aggregated by connection type. See also the description of Accounting field ADNOTACC.

Note: In contrast to ADNOTACC, unaccounted times of parallel records are included in SDNOTACC.

Field Name: SDNOTACC

QUANTITY

The number of transactions aggregated for the connection type. See also the description of Accounting field QWACPCNT.

Field Name: SWACPCNT
Authorization Management

This topic shows detailed information about “Statistics - Authorization Management”.

There are three authorization caches, located in the EDM pool:
- Plan, one cache per plan
- Package, one per subsystem
- Routine, for stored procedures and user-defined functions, one per subsystem

Allied threads (CICS, IMS, TSO, batch) are checked for EXECUTE authority at plan level. The package, and routine authorization caches only check EXECUTE authority for distributed applications.

The size of the plan authorization cache is set at BIND time, with the parameter CACHESIZE. When this is not specified, the default is taken from the ZPARM AUTHCACHE.

Plans with GRANT EXECUTE to public need no authorization checking, and therefore no cache (CACHESIZE=0). Other plans need enough cache space to store all authorized user IDs, approximately eight bytes per user ID.

The size of the package authorization cache is determined by ZPARM CACHEPAC.

The size of the routine authorization cache is determined by ZPARM CACHERAC.

Statistics - Authorization Management

The field labels shown in the following sample layout of “Statistics - Authorization Management” are described in the following section.

<table>
<thead>
<tr>
<th>AUTHORIZATION MANAGEMENT</th>
<th>QUANTITY /SECOND /THREAD /COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>TOTAL AUTH ATTEMPTS</td>
<td>53.00 0.29 N/C 1.61</td>
</tr>
<tr>
<td>TOTAL AUTH SUCC</td>
<td>53.00 0.29 N/C 1.61</td>
</tr>
<tr>
<td>PLAN-AUTH SUCC-W/O CATALOG</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
<tr>
<td>PLAN-AUTH SUCC-PUB-W/O CAT</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
<tr>
<td>PKG-AUTH SUCC-W/O CATALOG</td>
<td>16.00 0.09 N/C 0.48</td>
</tr>
<tr>
<td>PKG-AUTH SUCC-PUB-W/O CAT</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
<tr>
<td>PKG-AUTH UNSUCC-CACHE</td>
<td>1.00 0.01 N/C 0.03</td>
</tr>
<tr>
<td>PKG CACHE OVERWRT - AUTH ID</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
<tr>
<td>PKG CACHE OVERWRT - ENTRY</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
<tr>
<td>RTN-AUTH SUCC-W/O CATALOG</td>
<td>12.00 0.07 N/C 0.36</td>
</tr>
<tr>
<td>RTN-AUTH SUCC-PUB-W/O CAT</td>
<td>12.00 0.07 N/C 0.36</td>
</tr>
<tr>
<td>RTN-AUTH UNSUCC-CACHE</td>
<td>4.00 0.02 N/C 0.12</td>
</tr>
<tr>
<td>RTN CACHE OVERWRT - AUTH ID</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
<tr>
<td>RTN CACHE OVERWRT - ENTRY</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
<tr>
<td>RTN CACHE - ENTRY NOT ADDED</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
</tbody>
</table>

TOTAL AUTH ATTEMPTS

The number of authorization checks performed for plans, packages, and stored procedures since DB2 was started. This includes successful and failed checks.

Field Name: QTAUCHK

TOTAL AUTH SUCC
The number of successful authorization checks performed on plans, packages, and stored procedures, since DB2 was started.

Field Name: QTAUSUC

PLAN-AUTH SUCC-W/O CATALOG

The number of successful authorization checks that do not use the DB2 catalog (including plan cache checks and public checks).

Background and Tuning Information

For transaction level security, ENABLE and DISABLE on BIND PACKAGE should be used to ensure adequate security. Granting execute authority on the plan to public should be adequate.

Field Name: QTAUCCH

PLAN-AUTH SUCC-PUB-W/O CAT

The number of successful authorization checks based on EXECUTE authority granted to PUBLIC.

Field Name: QTAUPUB

PKG-AUTH SUCC-W/O CATALOG

The number of successful package EXECUTE authorization checks without accessing the DB2 catalog.

Field Name: QTPACAUT

PKG-AUTH SUCC-PUB-W/O CAT

The number of successful package EXECUTE authorization checks without accessing the DB2 catalog. Package EXECUTE authority was granted to PUBLIC in the package authorization cache.

Field Name: QTPACPUB

PKG-AUTH UNSUCC-CACHE

The number of unsuccessful package EXECUTE authorization checks in the package authorization cache. No applicable entry was found in the cache and DB2 catalog access was used.

Background and Tuning Information

This field indicates the efficiency of the cache. It should be close to zero.

Field Name: QTPACNOT

PKG CACHE OVERWRT - AUTH ID

The number of times an authorization ID was overwritten to add another one to the package authorization cache.

Field Name: QTPACOW1

PKG CACHE OVERWRT - ENTRY

The number of times an entry for a collection-ID or package-ID was overwritten to add another one to the package authorization cache.

Field Name: QTPACOW2

RTN-AUTH SUCC-W/O CATALOG

The number of times the routine authorization cache was checked successfully of EXECUTE authority on a stored procedure or user-defined
function. The DB2 catalog was not accessed. This counter includes the number of PUBLIC authorization checks.

**Field Name:** QTRACAUT

**RTN-AUTH SUCC-PUB-W/O CAT**

Number of successful authorization checks for user-defined function or stored procedure execution authority when that authority is held by PUBLIC. The DB2 catalog was not checked.

**Field Name:** QTRACPU

**RTN-AUTH UNSUCC-CACHE**

Number of unsuccessful authorization checks for user-defined function or stored procedure EXECUTE authority because no applicable entry was found in the routine authorization cache.

**Background and Tuning Information**

This field indicates the efficiency of the cache. It should be close to zero.

**Field Name:** QTRACNOT

**RTN CACHE OVERWRT - AUTH ID**

Number of times that DB2 overwrote an authorization ID in the routine authorization cache.

**Field Name:** QTRACOW1

**RTN CACHE OVERWRT - ENTRY**

Number of times that DB2 overwrote a routine entry in the routine authorization cache.

An entry in the routine authorization cache can refer to a function or procedure or to all functions or procedures within a specific schema.

**Field Name:** QTRACOW2

**RTN CACHE - ENTRY NOT ADDED**

Number of times that DB2 could not add an entry to the routine authorization cache.

An entry in the routine authorization cache can refer to a function or procedure or to all functions or procedures within a specific schema.

**Field Name:** QTRACNAC
Buffer Pool General

This topic shows detailed information about “Statistics - Buffer Pool General”.

This block is only printed when the buffer pool is active. If more than one 4 KB or 32 KB buffer pool block is present, a summary block showing buffer pool totals is also printed. If the report contains both 4 KB and 32 KB buffer pool blocks, a block showing the totals for all buffer pools is printed.

Statistics - Buffer Pool General

The field labels shown in the following sample layout of “Statistics - Buffer Pool General” are described in the following section.

<table>
<thead>
<tr>
<th>Field Label</th>
<th>Quantity</th>
<th>/Second</th>
<th>/Thread</th>
<th>/Commit</th>
</tr>
</thead>
<tbody>
<tr>
<td>CURRENT ACTIVE BUFFERS</td>
<td>164.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>UNAVAIL.BUFFER-VPOOL FULL</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>NUMBER OF DATASET OPENS</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>BUFFERS ALLOCATED - VPOOL</td>
<td>500.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>DFHSM MIGRATED DATASET</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DFHSM RECALL TIMEOUTS</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>VPOOL EXPANS. OR CONTRACT.</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>VPOOL EXPANS. FAILURES</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>CONCUR.PREF./I/O STREAMS-HWM</td>
<td>0.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>PREF./I/O STREAMS REDUCTION</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>PARALLEL QUERY REQUESTS</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>PARALLEL QUERY REQ.REDUCTION</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>PREF.QUANT.REDUCED TO 1/2</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>PREF.QUANT.REDUCED TO 1/4</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>NUMBER OF LPL INSERTS</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>MIN BUFFERS ON SLRU</td>
<td>0.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>MAX BUFFERS ON SLRU</td>
<td>0.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>SLRU LENGTH EQUALS VPSEQT</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>GETPAGE REQ RANDOM ON SLRU</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>

CURRENT ACTIVE BUFFERS

The total number of currently active (nonstealable) buffers. This field is an instantaneous sample of the number of buffers in the buffer pool that were updated or in use at the time this monitor data was requested. Because this field gives a snapshot value at statistics collection time, it only shows a problem if it happens at this time.

Background and Tuning Information

The buffer pool might be too small if the percentage of active pages in the buffer pool is beyond the deferred write threshold (DWQTH).

Field Name: QBSTCBA

UNAVAIL.BUFFER-VPOOL FULL

The number of times a usable buffer could not be located in the virtual buffer pool because the virtual buffer pool was full.

Background and Tuning Information
Buffer Pool General

Ideally, this value should be 0. Any other value indicates that the buffer pool is underallocated. In this case, use the ALTER BUFFERPOOL command to increase the virtual buffer pool size until this value remains at 0.

Field Name: QBSTXFL

NUMBER OF DATASET OPENS

The number of data sets physically opened successfully. This value is cumulative from the start of the DB2 statistics interval.

Field Name: QBSTDSO

BUFFERS ALLOCATED - VPOOL

The number of buffers allocated for a virtual buffer pool.

The number of buffers within each pool is always less than or equal to the corresponding value specified at installation time or when using the ALTER BUFFERPOOL command.

Background and Tuning Information

You should monitor the buffer pool hit ratio field to find the optimum size of the buffer pool. Usually the buffer pool hit ratio is improved by increasing the size of the buffer pool. However, paging the buffer pool storage impacts DB2 performance if the virtual buffer pool is too large.

Page-ins Required for Read I/O (QBSTRPI) and Page-ins Required for Write I/O (QBSTWPI) are useful when determining whether paging affects the performance of a certain buffer pool. The Resource Measurement Facility (RMF) also provides reports on MVS paging activity:

Storage Paging

When the virtual buffer pool is extended into expanded storage, MVS storage paging activity occurs. If a large buffer pool size results in excessive storage paging, consider using hiperpools.

Paging to Auxiliary Storage

If the virtual buffer pool size requirements exceed the central storage and expanded storage available, the oldest buffer pool pages migrate to auxiliary paging storage. When these pages are accessed subsequently, I/O must bring them back into real storage. This should be avoided. You could have a smaller buffer pool and let DB2 do the I/O rather than use MVS paging with its I/O CPU overhead. This is a situation that you (as the system programmer) should monitor.

You can use the ALTER BUFFERPOOL command to alter the size of the virtual buffer pool. However, the original buffer pool attributes reappear when DB2 stops and restarts.

Changing the size of the virtual buffer pool implicitly changes the buffer pool thresholds. See the Deferred Write Threshold Reached field (QBSTDWT).

Field Name: QBSTVPL

DFHSM MIGRATED DATASET

The number of times migrated data sets were encountered.

Field Name: QBSTMIG
DFHSM RECALL TIMEOUTS
The number of recall timeouts.
Field Name: QBSTRTO

VPOOL EXPANS. FAILURES
The total number of virtual buffer pool expansion failures due to the lack of virtual storage space.

Background and Tuning Information
Ideally, this value should be 0. If it is not, check the virtual storage allocation of the DB2 database address space for areas that can be reduced. For example, you can reduce the size of other buffer pools.
Field Name: QBSTXVF
This is an exception field.

CONCUR.PREF.I/O STREAMS-HWM
The highest number of concurrent prefetch I/O streams allocated to support a parallel I/O or CP query in this buffer pool. It reflects prefetch activities for non-workfile page sets.
This number only applies to query I/O and CP parallelism.
Field Name: QBSTXIS
This is an exception field.

PREF.I/O STREAMS REDUCTION
The total number of requested prefetch I/O streams that were denied because of a lack of buffer pool storage space.
It only applies to query I/O and CP parallelism.
For example, if 100 prefetch I/O streams are requested and only 80 are granted, then 20 is added to the number in this field.

Background and Tuning Information
Consider increasing the size of the buffer pool if this value is not 0.
The ratio of this field and the Reduced parallel query requests field gives the average degree of parallel query processing that was reduced because of insufficient buffer pool space. The Prefetch I/O streams - Concurrent streams - high-water mark field gives the highest degree of parallel query processing that was reduced for one or more queries processed in parallel.
The number in this field reflects the prefetch activities for non-workfile page sets.
Field Name: QBSTJIS
This is an exception field.

PARALLEL QUERY REQUESTS
The total number of requests made for parallel query support in this buffer pool. This field only applies to non-workfile page sets in query I/O and CP parallelism.
Field Name: QBSTPQO

PARALL.QUERY REQ.REDUCTION
Buffer Pool General

The number of times that DB2 could not allocate the requested number of buffer pages to allow a parallel group to run as planned.

This field only applies to non-workfile page sets in query I/O and CP parallelism.

Background and Tuning Information

This is caused by a shortage of storage in the buffer pool. A nonzero value could suggest that the buffer pool is too small. You can increase it using the ALTER BUFFERPOOL command.

Field Name: QBSTPQF

This is an *exception* field.

**PREF.QUANT.REDUCED TO 1/2**

The total number of times prefetch quantity is reduced from normal to 50% of normal. The normal size depends on the page size of the buffer pool.

This field only applies to query I/O and CP parallelism.

Background and Tuning Information

The number in this field indicates when DB2 had to reduce the sequential prefetch quantity to continue executing concurrently with parallel queries in the system. If the number is small, it may be tolerable.

Field Name: QBSTPL1

This is an *exception* field.

**PREF.QUANT.REDUCED TO 1/4**

The total number of times prefetch quantity is reduced from 50% to 25% of normal. The normal size depends on the page size of the buffer pool.

This field only applies to query I/O and CP parallelism.

Background and Tuning Information

The query response for parallel queries can be significantly degraded if the value in this field is not 0.

Field Name: QBSTPL2

This is an *exception* field.

**NUMBER OF LPL INSERTS**

The number of times that one or more pages were added to the logical page list (LPL).

Field Name: QBSTLPL

**MIN BUFFERS ON SLRU**

The minimum number of buffers on the sequential least-recently-used (SLRU) chain in the last statistical period. This is the low-water mark (LWM) within an interval.

Field Name: QBSTSMIN

**MAX BUFFERS ON SLRU**

The maximum number of buffers on the sequential least-recently-used (SLRU) chain in the last statistical period. This is the high-water mark (HWM) within an interval.
Field Name: QBSTSMAX

SLRU LENGTH EQUALS VPSEQT

The number of times when the length of the sequential least-recently-used (SLRU) chain equals the sequential steal threshold VPSEQT.

Field Name: QBSTHST

GETPAGE REQU RANDOM ON SLRU

The number of times that the random Getpage request has a buffer hit and the buffer is on the least-recently-used (SLRU) chain.

Field Name: QBSTRHS
Buffer Pool Read

This topic shows detailed information about “Statistics - Buffer Pool Read”.

This block is only printed when the buffer pool is active. If more than one 4 KB or 32 KB buffer pool block is present, a summary block showing buffer pool totals is also printed. If the report contains both 4 KB and 32 KB buffer pool blocks, a block showing the totals for all buffer pools is printed.

Statistics - Buffer Pool Read

The field labels shown in the following sample layout of “Statistics - Buffer Pool Read” are described in the following section.

<table>
<thead>
<tr>
<th>BP0 READ OPERATIONS</th>
<th>QUANTITY /SECOND /THREAD /COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>BPOOL HIT RATIO (%)</td>
<td>100.00</td>
</tr>
<tr>
<td>BPOOL HIT RATIO (%) SEQU</td>
<td>100.00</td>
</tr>
<tr>
<td>BPOOL HIT RATIO (%) RANDOM</td>
<td>100.00</td>
</tr>
<tr>
<td>GETPAGE REQUEST</td>
<td>30390.00</td>
</tr>
<tr>
<td>GETPAGE REQUEST-SEQUENTIAL</td>
<td>8541.00</td>
</tr>
<tr>
<td>GETPAGE REQUEST-RANDOM</td>
<td>21849.00</td>
</tr>
<tr>
<td>SYNCHRONOUS READS</td>
<td>5013.00</td>
</tr>
<tr>
<td>SYNCHRON. READS-SEQUENTIAL</td>
<td>0.00</td>
</tr>
<tr>
<td>SYNCHRON. READS-RANDOM</td>
<td>0.00</td>
</tr>
<tr>
<td>GETPAGE PER SYN.READ-RANDOM</td>
<td>N/C</td>
</tr>
<tr>
<td>SEQUENTIAL PREFETCH REQUEST</td>
<td>0.00</td>
</tr>
<tr>
<td>SEQUENTIAL PREFETCH READS</td>
<td>0.00</td>
</tr>
<tr>
<td>PAGES READ VIA SEQ.PREFETCH</td>
<td>0.00</td>
</tr>
<tr>
<td>S.PRF.PAGES READ/S.PRF.READ</td>
<td>N/C</td>
</tr>
<tr>
<td>LIST PREFETCH REQUESTS</td>
<td>0.00</td>
</tr>
<tr>
<td>LIST PREFETCH READS</td>
<td>0.00</td>
</tr>
<tr>
<td>PAGES READ VIA LIST PREFETCH</td>
<td>0.00</td>
</tr>
<tr>
<td>L.PRF.PAGES READ/L.PRF.READ</td>
<td>N/C</td>
</tr>
<tr>
<td>DYNAMIC PREFETCH REQUESTED</td>
<td>15.00</td>
</tr>
<tr>
<td>DYNAMIC PREFETCH READS</td>
<td>0.00</td>
</tr>
<tr>
<td>PAGES READ VIA DYN.PREFETCH</td>
<td>0.00</td>
</tr>
<tr>
<td>D.PRF.PAGES READ/D.PRF.READ</td>
<td>N/C</td>
</tr>
<tr>
<td>PREF.DISABLED-NO BUFFER</td>
<td>0.00</td>
</tr>
<tr>
<td>PREF.DISABLED-NO READ ENG</td>
<td>0.00</td>
</tr>
<tr>
<td>PAGE-INS REQUIRED FOR READ</td>
<td>0.00</td>
</tr>
</tbody>
</table>

BPOOL HIT RATIO (%)

The percentage of Getpage operations that were satisfied by a page already in the buffer pool.

The value is calculated as the ratio of number of successful Getpage operations minus the number of pages read from DASD (both synchronously and using prefetch), to the number of successful Getpage operations, expressed as a percentage.

Background and Tuning Information

The highest possible hit ratio is 100%, that is, when every page requested is always in the buffer pool. If the requested page is not in the buffer pool, the hit ratio is 0% or less. If the hit ratio is negative, this means that prefetch brought pages into the buffer pool that are not subsequently referenced, either because the query stops before it reaches the end of the table space, or because the prefetched pages are stolen by DB2 for reuse before the query can access them. A low buffer pool hit ratio is not necessarily bad. The hit ratio is a relative value, based on the type of application. For example, an application that browses large data might
Buffer Pool Read

have a buffer pool hit ratio of 0. Watch for those cases where the hit ratio drops significantly for the same application. Here are some suggestions to increase the buffer hit ratio:

• Run the REORG utility for indexes or table spaces associated with the virtual buffer pool.
• Reserve more pages for random I/O by setting the SEQUENTIAL STEAL THRESHOLD (VPSEQT) to a lower value.
• Increase the buffer pool as long as the cost of paging does not outweigh the benefit of I/O avoidance.
• Establish more separate buffer pools, perhaps to isolate different applications.

When the hit ratio is negative, it means that prefetch has brought pages into the buffer pool that are not subsequently referenced, either because the query stops before it reaches the end of the table space, or because the prefetched pages are stolen by DB2 for reuse before the query can access them.

The hit ratio measurement becomes less meaningful if the buffer pool is used by additional processes, such as utilities or work files.

Field Name: SBUFRAT

**BPOOL HIT RATIO (%) SEQU**
The percentage of sequential Getpage operations that were satisfied by a page already in the buffer pool.

Field Name: SBUFFSEQ

**BPOOL HIT RATIO (%) RANDOM**
The percentage of random Getpage operations that were satisfied by a page already in the buffer pool. If this value is low, it indicates that page residency in the buffer pool is too low, therefore the buffer pool may be too small.

Field Name: SBUFRDM

**GETPAGE REQUEST**
The number of Getpage requests including conditional and unconditional requests.

Field Name: QBSTGET

**GETPAGE REQUEST-SEQUENTIAL**
The number of Getpage requests issued by sequential access requesters.

Field Name: QBSTSGT

**GETPAGE REQUEST-RANDOM**
The number of random Getpage requests.

Field Name: SDGETRAN

**SYNCHRONOUS READS**
The number of synchronous read I/O operations performed by DB2 for applications and utilities.

**Background and Tuning Information**
Buffer Pool Read

This number includes both Synchronous Reads Sequential Access Only (QBSTSIO) and synchronous read operations for non-sequential access.

You can use this value and the value of Synchronous Reads Sequential Access Only to calculate the number of Non-Sequential Synchronous Reads.

Check the buffer pool hit ratio if the number of non-sequential synchronous reads is larger than expected.

Field Name: QBSTRIO

This is an exception field.

SYNCHRON. READS-SEQUENTIAL

The number of synchronous read I/O requests issued by sequential access requesters.

Background and Tuning Information

Sequential synchronous read I/Os can occur because:

- Prefetch is disabled (QBSTSPD).
- Prefetch pages could have been stolen from the buffer pool before the Getpage request is issued for those pages. Subsequently the pages are reread synchronously. A negative buffer pool hit ratio can indicate the same problem.
- The pages requested are not consecutive: DB2 estimated the selected range of pages to be so small that prefetch would make no sense. See also Sequential Prefetch Requested (QBSTSEQ).

It is normal to have a small value for SYNC READ I/O (SEQUENTIAL) because before the sequential prefetch is scheduled, the first page of a prefetch is read by SYNC READ I/O. However, if this number is large, consider increasing the size of the buffer pool or reviewing the sequential steal thresholds (VPSEQT and HPSEQT).

Field Name: QBSTSIO

This is an exception field.

SYNCHRON. READS-RANDOM

The number of random synchronous read I/O requests.

Field Name: SDSTRAN

This is an exception field.

GETPAGE PER SYN.READ-RANDOM

The number of random Getpage requests per random synchronous read I/O request.

Background and Tuning Information

This ratio is a good indicator of read efficiency in a transaction environment. The higher the number is, the better.

Field Name: SBRGPRIO

SEQUENTIAL PREFETCH REQUEST

The number of sequential prefetch requests. This counter is incremented for each PREFETCH request (which can result in an I/O read). If the prefetch results in an I/O read, up to 32 pages may be read for SQL, and
Buffer Pool Read

up to 64 pages for utilities. A request does not result in an I/O read if all pages to be prefetched are already in the buffer pool.

This counter does not include sequential detection, which is recorded in the Dynamic Prefetch - Requested field.

Background and Tuning Information

Sequential prefetch reads a sequential set of pages. It allows CP and I/O operations to be overlapped. DB2 determines at BIND time whether sequential prefetch is used or not.

Sequential prefetch is generally used for a table space scan. It can also be used to read index pages in an index scan. For an index scan that accesses 8 or more consecutive data pages, DB2 requests sequential prefetch at bind time. The index must have a cluster ratio of 80% or higher. (Use REORG and RUNSTATS and rebind relevant SQL if you doubt that this target has been met previously.)

The number of prefetch requests by itself is not a good indicator for efficiency of prefetching:

• At run time not every prefetch request results in read I/O: the Sequential Prefetch Reads field (QBSTPIO) shows the number of read I/O operations caused by sequential prefetch. The Prefetch Disabled No Buffer (QBSTSPD) and Prefetch Disabled No Read Engine fields (QBSTREE) show the number of times prefetch was disabled because buffers and read engines had not been available.

• Check the value in the buffer pool hit ratio. A negative value indicates that prefetched pages are stolen from the buffer pool before they are read. The pages are subsequently reread synchronously. There will be also a large value in the Synchronous Reads Total (QBSTRIO) field.

• Decreasing the size of the buffer pool can reduce the prefetch quantity, leading to a larger number of prefetch requests. See also the Sequential Prefetch Pages Read field (QBSTSPP).

Field Name: QBSTSEQ

This is an exception field.

SEQUENTIAL PREFETCH READS

The number of asynchronous read I/O operations due to normal sequential prefetch (applications and utilities).

Background and Tuning Information

Prefetch Read I/O is not activated if one of the following conditions applies:

• All pages in the prefetch range are already in the buffer pool.

• Prefetch is disabled (QBSTSPD).

This means that the value in this field is usually smaller than the number of sequential prefetch requests (QBSTSEQ).

Field Name: QBSTPIO

This is an exception field.

PAGES READ VIA SEQ.PREFETCH

The total number of pages read due to a normal sequential prefetch. A sequential prefetch request does not result in a read I/O if all the desired pages are found in the buffer pool.
Buffer Pool Read

Background and Tuning Information

The ratio of Sequential Prefetch Pages Read to Sequential Prefetch Reads (QBSTPIO) is usually between 0 and 32.

For requests issued by application programs, the number of pages per READ I/O primarily depends on the page size and the size of the buffer pool. Normally thirty-two 4 KB pages (or four 32 KB pages) is the maximum prefetch quantity for table space scans, whether data or index. Utilities use a prefetch quantity of up to 64 pages.

The number of pages per READ I/O can be lower because:
- Pages within the prefetch range may already be in the buffer pool.
- Not enough pages are available because of a buffer shortage.
- A prefetch quantity of 8 pages or less is used for work files.

A small value for this ratio can indicate:
- A good performing buffer pool being so large that most of the pages, which had otherwise to be prefetched, are cached in the buffer pool. In this case, the buffer pool hit ratio should be high.
- A buffer shortage condition, reducing the efficiency of sequential prefetch. This could mean, for example, work-file prefetch quantity reduction from 8 to 4 to 2, as the number of available buffers shrinks. In this case, you should consider tuning the buffer pool.

Field Name: QBSTSP
This is an exception field.

S.PRF.PAGES READ/S.PRF.READ
The number of sequential prefetch pages read per sequential prefetch read I/O operation.

Field Name: SBRRPPRIO
This is an exception field.

LIST PREFETCH REQUESTS
The number of list prefetch requests.

List prefetch allows DB2 to access data pages efficiently even when the required data pages are not contiguous. It allows CP and I/O operations to be overlapped.

Background and Tuning Information

DB2 determines at BIND time whether sequential prefetch is used. List prefetch is chosen as follows:
- Usually with a single index that has a cluster ratio lower than 80%.
- Sometimes on a single index with a high cluster ratio, if the estimated amount of data to be accessed is too small to make sequential prefetch efficient.
- Always to access data by multiple index access.
- Always to access data from the inner table during a hybrid join.

DB2 never chooses list prefetch if the estimated number of RIDs to be processed takes more than 50% of the RID pool. During execution time, list prefetch processing terminates if more than 25% of the rows (with a minimum of 4075) in the table must be accessed.
Data pages are read in quantities equal to the sequential prefetch quantity (QBSTSEQ), which depends on buffer pool size and is usually 32 pages.

**Field Name:** QBSTLPF

This is an *exception* field.

**LIST PREFETCH READS**

The number of asynchronous read I/O operations caused by the list prefetch.

The number of pages read is shown by the List Prefetch Pages Read (QBSTLPP) field.

**Background and Tuning Information**

Prefetch Read I/O is not activated if one of the following conditions apply:

- All pages in the prefetch range are already in the buffer pool.
- Prefetch is disabled (Prefetch Disabled No Read Engine - QBSTREE).

This means that the value in this field is usually less than the number of list prefetch requests (QBSTLPF).

**Field Name:** QBSTLIO

This is an *exception* field.

**PAGES READ VIA LIST PREFETCH**

The number of pages read via list prefetch.

**Field Name:** QBSTLPP

**L.PRF.PAGES READ/L.PRF.READ**

The number of list prefetch pages read per list prefetch read I/O.

**Field Name:** SDLPPPPIO

This is an *exception* field.

**DYNAMIC PREFETCH REQUESTED**

The number of dynamic prefetch requests. Dynamic prefetch is the process that is triggered because of sequential detection. If the prefetch request results in an I/O read, up to 32 advancing pages can be read at a time.

**Background and Tuning Information**

Dynamic prefetch reads a sequential set of pages. It allows CP and I/O operations to be overlapped. If DB2 does not choose prefetch at bind time it can sometimes use it at execution time. The method is called sequential detection.

The number of prefetch requests by itself is not a good indicator for efficiency of prefetching because:

- At run time not every prefetch request results in read I/O: the Dynamic Prefetch Reads field shows the number of read I/O operations caused by dynamic prefetch. The Prefetch Disabled No Buffer (QBSTSPD) and Prefetch Disabled No Read Engine (QBSTREE) fields show the number of times prefetch was disabled because buffers and read engines had not been available.
- Prefetch pages can be stolen from the buffer pool before they are read. This is indicated by a negative buffer pool hit ratio. The pages are...
Buffer Pool Read

subsequently reread synchronously. This will also cause an unexpectedly large value for total synchronous reads (QBSTRIO).

Decreasing the size of the buffer pool can reduce the prefetch quantity (QBSTDPP), leading to a larger number of prefetch requests.

Field Name: QBSTDPP
This is an exception field.

DYNAMIC PREFETCH READS
The number of asynchronous read I/Os because of dynamic prefetch. The number of pages read is recorded in the Dynamic Prefetch Pages Read field.

Background and Tuning Information
A prefetch request does not result in an I/O if one of the following conditions apply:
• All pages to be prefetched are already in the buffer pool.
• The prefetch is canceled.
This means that the value in this field is usually smaller than the number of dynamic prefetch requests.

Field Name: QBSTDIO
This is an exception field.

PAGES READ VIA DYN.PREFETCH
The number of pages read because of dynamic prefetch. Dynamic prefetch is the process that is triggered by sequential detection.

Background and Tuning Information
The ratio of Dynamic Prefetch Pages Read to Dynamic Prefetch Reads is between 0 and 32.
DB2 can fetch up to 32 pages per prefetch.
The number of pages per READ I/O can be lower because:
• Pages within the prefetch range are already in the buffer pool.
• Not as many pages are available due to a buffer shortage.
A small value for this ratio can indicate:
• A good performing buffer pool being large enough to contain pages that would otherwise be prefetched. This is indicated by a high buffer pool hit ratio.
• A buffer shortage condition, which reduces the efficiency of dynamic prefetch. In this instance the buffer pool hit ratio will be low. Consider tuning the buffer pool.

Field Name: QBSTDPP
This is an exception field.

D.PRF.PAGES READ/D.PRF.READ
The number of dynamic prefetch pages read per dynamic prefetch read I/O.

Field Name: SDDPPPIO
This is an exception field.
Buffer Pool Read

**PREF.DISABLED-NO BUFFER**

The total number of times sequential prefetch was disabled because buffers were not available.

*Field Name*: QBSTSPD

This is an *exception* field.

**PREF.DISABLED-NO READ ENG**

The total number of times a prefetch is disabled because of an unavailable read engine.

**Background and Tuning Information**

Because there are 600 read engines, a maximum of 600 concurrent prefetch operations can be processed at a time. When this maximum is reached, prefetching is disabled and this count is incremented. The value in this field should be close to 0.

*Field Name*: QBSTREE

This is an *exception* field.

**PAGE-INS REQUIRED FOR READ**

The number of page-ins required for a read I/O.

*Field Name*: QBSTRPI

This is an *exception* field.
Buffer Pool Sort/Merge

This topic shows detailed information about “Statistics - Buffer Pool Sort/Merge”.

This block is only printed when the buffer pool is active. If more than one 4 KB or 32 KB buffer pool block is present, a summary block showing buffer pool totals is also printed. If the report contains both 4 KB and 32 KB buffer pool blocks, a block showing the totals for all buffer pools is printed.

Statistics - Buffer Pool Sort/Merge

The field labels shown in the following sample layout of “Statistics - Buffer Pool Sort/Merge” are described in the following section.

<table>
<thead>
<tr>
<th>MAX WORKFILES CONCURR. USED</th>
<th>QUANTITY /SECOND /THREAD /COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>MERGE PASSES REQUESTED</td>
<td>0.00 N/A N/A N/A</td>
</tr>
<tr>
<td>MERGE PASS DEGRADED-LOW BUF</td>
<td>0.00 0.00 N/C N/C</td>
</tr>
<tr>
<td>WORKFILE REQ.REJCTD-LOW BUF</td>
<td>0.00 0.00 N/C N/C</td>
</tr>
<tr>
<td>WORKFILE REQ-ALL MERGE PASS</td>
<td>0.00 0.00 N/C N/C</td>
</tr>
<tr>
<td>WORKFILE PRF NOT SCHEDULED</td>
<td>0.00 0.00 N/C N/C</td>
</tr>
<tr>
<td>MAX WORKFILES CONCURR. USED</td>
<td>0.00 N/A N/A N/A</td>
</tr>
</tbody>
</table>

MAX WORKFILES CONCURR. USED

The maximum number of work files concurrently used during merge processing within this statistics period.

Ideally, each work file needs 16 buffers to allow DB2 to perform a sequential prefetch for work files.

Field Name: QBSTWFM

This is an exception field.

MERGE PASSES REQUESTED

The total number of merge passes for DB2 sort activities. This value reflects how many merge passes were requested for DB2 to determine the number of work files permitted to support each merge pass.

Field Name: QBSTWFR

MERGE PASS DEGRADED-LOW BUF

The number of times that a merge pass was not efficiently performed due to a shortage of space in the buffer pool. The number in this field is incremented for each merge pass where the maximum number of work files allowed is less than the number of work-files requested.

Background and Tuning Information

The maximum number of work files allowed is calculated as follows:

• Buffers consumed = 2 * (work files already allocated)
• Buffers available = (sequential steal threshold * buffer pool size - buffers consumed)
• Maximum work files allowed = buffers available / (2 * 8)

The default for the sequential steal threshold is 0.8.

Ideally, the number in this field should be 0. Otherwise, it indicates a shortage of buffer pool space or that there are too many concurrent work
files. For example, there could be a number of concurrently open cursors that require sorting. Consider increasing the buffer pool size using the ALTER BUFFERPOOL command.

**Field Name:** QBSTWFF

This is an *exception* field.

**WORKFILE REQ.REJCTD-LOW BUF**

The total number of work files that were rejected during all merge passes because of insufficient buffer resources.

**Background and Tuning Information**

This field and the degraded low buffers field determine the average number of work files that cannot be honored at each merge pass because of insufficient buffer pool space.

Ideally, the number in this field should be 0. Otherwise, it indicates a shortage of buffer pool space or that there are too many concurrent work files. For example, there could be a number of concurrently open cursors that require sorting. Consider increasing the size of the buffer pool using the ALTER BUFFERPOOL command.

Note that, when there are many concurrent sorts or large sorts, it is a good idea to dedicate a separate buffer pool for sort work files. This will greatly facilitate work-file performance tuning.

**Field Name:** QBSTWFD

This is an *exception* field.

**WORKFILE REQ-ALL MERGE PASS**

The total number of work files requested for all merge passes.

This field and the Merge Passes Requested field determine the average number of work files requested in a single merge pass.

For DB2 to perform an efficient prefetch for work files, each workfile should have at least 16 dedicated buffers. Work files used during sort phase processing or other non-sort-related processing are not included in this number.

**Field Name:** QBSTWFT

**WORKFILE NOT CREATED-NO BUF**

This field is only applicable if DB2 is running under MVS/XA.

The number of times a work file could not be created due to insufficient buffer resources. It indicates that a sort is in progress and limited in regard to the number of work files it can use.

**Background and Tuning Information**

Ideally, this should be 0. Otherwise, it indicates a shortage of buffer pool space or that there are many concurrent work files. For example, there could be a number of open cursors that require sorting.

Generally, sorts are performed more efficiently with additional work files, but there are internal DB2 limits on the number of work files a transaction can have. It is possible that at run time a transaction cannot use as many
Buffer Pool Sort/Merge

work files as it had planned. You can control this by increasing the buffer pool size (ALTER BUFFERPOOL), or changing the transaction so it requires fewer concurrent work files.

**Field Name:** QBSTMAX

This is an *exception* field.

**WORKFILE PRF NOT SCHEDULED**

The number of times a sequential prefetch was not scheduled for a work file because the dynamic prefetch quantity is zero.

**Background and Tuning Information**

The work-file prefetch checks the dynamic prefetch quantity (normally 1 to 8 pages). When the quantity is zero, the value in this field is incremented. A high number in this field implies that the buffer pool is too small.

Ideally, the number in this field should be 0. Otherwise, it indicates a shortage of buffer pool space or that there are many concurrent work files. For example, there could be a number of concurrently open cursors that require sorting.

Consider increasing the size of the buffer pool or allocating a buffer pool specifically for DSND807 usage. This can be especially effective with high-use query systems whose reports make extensive use of sort activity.

**Field Name:** QBSTWKPD

This is an *exception* field.
Buffer Pool Write

This topic shows detailed information about “Statistics - Buffer Pool Write”.

This block is only printed when the buffer pool is active. If more than one 4 KB or 32 KB buffer pool block is present, a summary block showing buffer pool totals is also printed. If the report contains both 4 KB and 32 KB buffer pool blocks, then a block showing the totals for all buffer pools is printed.

Statistics - Buffer Pool Write

The field labels shown in the following sample layout of “Statistics - Buffer Pool Write” are described in the following section.

<table>
<thead>
<tr>
<th>BPO WRITE OPERATIONS</th>
<th>QUANTITY /SECOND /THREAD /COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>BUFFER UPDATES</td>
<td>6257.00 34.42 N/C N/C</td>
</tr>
<tr>
<td>PAGES WRITTEN</td>
<td>939.00 5.17 N/C N/C</td>
</tr>
<tr>
<td>BUFF_UPDATES/PAGES WRITTEN</td>
<td>6.66</td>
</tr>
<tr>
<td>SYNCHRONOUS WRITES</td>
<td>75.00 0.41 N/C N/C</td>
</tr>
<tr>
<td>ASYNCHRONOUS WRITES</td>
<td>282.00 1.55 N/C N/C</td>
</tr>
<tr>
<td>PAGES WRITEN PER WRITE I/O</td>
<td>2.63</td>
</tr>
<tr>
<td>PAGES WRITN FOR CASTOUT I/O</td>
<td>0.00</td>
</tr>
<tr>
<td>NUMBER OF CASTOUT I/O</td>
<td>0.00 0.00 N/C N/C</td>
</tr>
<tr>
<td>HORIZ.DEF.WRITE THRESHOLD</td>
<td>0.00 0.00 N/C N/C</td>
</tr>
<tr>
<td>VERTI.DEF.WRITE THRESHOLD</td>
<td>0.00 0.00 N/C N/C</td>
</tr>
<tr>
<td>DM THRESHOLD</td>
<td>0.00 0.00 N/C N/C</td>
</tr>
<tr>
<td>PAGE-INS REQUIRED FOR WRITE</td>
<td>0.00 0.00 N/C N/C</td>
</tr>
</tbody>
</table>

BUFFER UPDATES

The number of times buffer updates were requested against pages in the buffer pool.

Background and Tuning Information

The ratio of Buffer Updates to Pages Written (QBSTPWS) suggests a high level of efficiency as it increases, because more updates are being externalized per physical write.

Buffer updates per pages written depends strongly on the type of application. For example, a batch program that processes a table in skip sequential mode with a high row update frequency in a dedicated environment can achieve very good update efficiency. In contrast, update efficiency tends to be lower for transaction processing applications, because transaction processing tends to be random.

The following can influence the number of updates per page:

Number of rows per page

A small PCTFREE value gathers more rows on the same page. However, at the same time this can impact concurrency.

Buffer pool size and deferred write thresholds

Increase DWQT and VDWQT or the size of the buffer pool. This causes DB2 to let page updates accumulate in the buffer pool. Therefore, the probability that more updates per page get captured increases. This effect is less significant if the buffer pool is concurrently used by several transactions, it also depends on the type of transaction.

Field Name: QBSTSWS
Buffer Pool Write

This is an exception field.

PAGES WRITTEN

The number of pages in the buffer pool written to DASD.

Background and Tuning Information

Consider the ratio of Pages Written per write I/O. The number of write I/O operations includes Asynchronous Writes (QBSTWIO) and Synchronous Writes (QBSTIMW).

The ratio of pages per write I/O suggests a high level of efficiency as the ratio increases, because more pages are being externalized per physical write.

The following factors impact the ratio of pages written per write I/O:

Checkpoint frequency

At checkpoint time, I/Os are scheduled to write all updated pages on the deferred write queue to DASD. If this occurs too frequently, the deferred write queue does not grow large enough to achieve a high ratio of pages written per write I/O.

The checkpoint frequency depends on the number of logs written between two consecutive checkpoints. This number is set at installation time; see the field CHECKPOINT FREQ of installation panel DSNTIPN.

Frequency of active log switch

DB2 takes a system checkpoint each time the active log is switched. High frequency of active log switches causes the problem described under checkpoint frequency.

Buffer pool size and deferred write thresholds

The deferred write thresholds (VDWQT and DWQT) are a function of buffer pool size. If the buffer pool size is decreased, these thresholds are reached more frequently, causing I/Os to be scheduled more often to write some of the pages on the deferred write queue to DASD. This prevents the deferred write queue from growing large enough to achieve a high ratio of pages written per write I/O.

Number of data sets, and the spread of updated pages across them

The efficiency of write I/O also depends on the number of data sets associated with the buffer pool and spread of updated pages across them. Because of the nature of batch processing, the ratio of pages written to write I/Os can be expected to be higher than that expected for transaction type workloads.

To determine update efficiency, use also the value in the Buffer Updates field (QBSTSW) to check the number of buffer updates per page written.

Field Name: QBSTPWS

This is an exception field.

BUFEUPDATES/PAGES WRITTEN

The number of buffer updates per page written from the buffer pool to DASD.
The ratio of BUFFER UPDATES (QBSTSWS) to PAGES WRITTEN (QBSTPWS) suggests a high level of efficiency as the ratio increases, because more updates are being externalized per physical write. For example, if there are 10 updates on the same page before it is externalized, then the ratio is 10:1 or 10. If all 10 updates are on 10 distinct pages, then the ratio is 10:10 or 1.

**Background and Tuning Information**

Buffer updates per pages written depends strongly on the type of application. For example, a batch program that processes a table in skip sequential mode with a high row update frequency in a dedicated environment can achieve very good update efficiency. In contrast, update efficiency tends to be lower for transaction processing applications, because transaction processing tends to be random.

The following factors can influence the number of updates per page:

- **Number of rows per page**
  - A small PCTFREE value will gather more rows on the same page. However, at the same time this can have impact on concurrency.

- **Buffer pool size and deferred write thresholds**
  - Increase DWQT and VDWQT or the size of the buffer pool. This would tell DB2 to let page updates accumulate in the buffer pool. This means, the probability that more updates per page get captured increases. This effect is less significant if the buffer pool is concurrently used by multiple transactions, it depends on the type of transaction.

**Field Name:** SBRBUPW

**SYNCHRONOUS WRITES**

The total number of immediate writes. Immediate writes occur when:

- An immediate write threshold is reached
- No deferred write engines are available
- More than two checkpoints pass without a page being written.

Sometimes DB2 uses synchronous writes even when the IWTH is not exceeded. As an example, when more than two checkpoints pass without a page being written. This type of situation does not indicate a buffer shortage.

**Background and Tuning Information**

A small number of immediate writes can be expected. Synchronous writes occur if there are too many checkpoints and/or the buffer pool is too small.

If a large number of synchronous writes occur, monitor the DM Critical Threshold Reached (QBSTDMC) field. Reaching Immediate Write Threshold (IWTH-97.5%) implies that the Data Management Threshold (DMTH-95%) has been crossed. You can ignore the value in the immediate write field when DM Critical Threshold Reached is zero. Otherwise consider increasing the size of the buffer pool. You can use the ALTER BUFFERPOOL command. However, the original buffer pool attributes reappear when DB2 stops and restarts.
Buffer Pool Write

Check also the System Event Checkpoint field (QWSDCKPT) in the Subsystem Services block to see whether the frequency of DB2 checkpoints should be reduced. To do this, increase the value of ZPARM LOGLOAD.

Field Name: QBSTIMW
This is an exception field.

ASYNCHRONOUS WRITES
The number of asynchronous write I/O operations performed by media manager to a direct access storage device.

Field Name: QBSTWIO
This is an exception field.

PAGES WRITTEN PER WRITE I/O
The number of pages written from the buffer pool to DASD per synchronous or asynchronous write I/O. This count does not include preformatting I/O, such as I/O needed to prepare a data set for use.

Background and Tuning Information
The following factors impact the ratio of pages written per write I/O:

Checkpoint frequency
At checkpoint time, I/Os are scheduled to write all updated pages on the deferred write queue to DASD. If this occurs too frequently, the deferred write queue does not grow large enough to achieve a high ratio of pages written per write I/O.

The checkpoint frequency depends on the number of logs written between two consecutive checkpoints. This number is set at installation time; see the field CHECKPOINT FREQ of installation panel DSNTIPN.

Frequency of active log switch
DB2 takes a system checkpoint each time the active log is switched. High frequency of active log switches causes the problem described under checkpoint frequency.

Buffer pool size and deferred write thresholds
The deferred write thresholds (VDWQT and DWQT) are a function of buffer pool size. If the buffer pool size is decreased, these thresholds are reached more frequently, causing I/Os to be scheduled more often to write some of the pages on the deferred write queue to DASD. This prevents the deferred write queue from growing large enough to achieve a high ratio of pages written per write I/O.

Number of data sets, and the spread of updated pages across them
The efficiency of write I/O also depends on the number of data sets associated with the buffer pool and spread of updated pages across them. Because of the nature of batch processing, the ratio of pages written to write I/Os can be expected to be higher than that expected for transaction type workloads.

To determine update efficiency check also the ratio Buffer Updates / Pages Written (SBRBUPW).

Field Name: SBRPWWIO
PAGES WRTN FOR CASTOUT I/O

The number of pages written for castout I/O operations.

Field Name: QBSTPCO

NUMBER OF CASTOUT I/O

The number of castout I/O operations.

Field Name: QBSTCIO

HORIZ.DEF.WRITE THRESHOLD

The number of times the deferred write threshold (DWTH) was reached.

This threshold is a percentage of the virtual buffer pool that might be occupied by unavailable pages, including both updated pages and pages in use. DB2 checks this threshold when an update to a page is completed. If the percentage of unavailable pages in the virtual buffer pool exceeds the threshold, write operations are scheduled for enough data sets (up to 128 pages per data set) to reduce the number of unavailable buffers to 10% below the threshold.

Background and Tuning Information

The default value for this threshold is 50%. You can change that to any value from 0% to 90% by using the DWQT option on the ALTER BUFFERPOOL command.

The deferred write thresholds, DWQT and VDWQT, are specified as a percentage, their absolute value depends on the size of the virtual buffer pool.

Consider the following aspects when changing the deferred write thresholds:

Optimize the ratio of pages written per write I/O

The ratio can be monitored using the Pages Written (QBSTPWS) field.

When the buffer pool is relatively small, the default thresholds could prevent the deferred write queue from growing large enough to achieve a high ratio of pages written per write I/O. Raising these thresholds will, in this instance, reduce the I/O write frequency, increasing the number of pages written per I/O.

Distribute I/O evenly over time

If a virtual buffer pool is very large, it is unlikely that the default values of either DWQT or VDWQT will ever be reached. In this case, write I/Os tend to occur in surges, triggered by DB2 checkpoints. Lowering the VDWQT and the DWQT could improve performance by distributing the write I/Os more evenly over time.

Impact on other buffer pool thresholds

Increasing DWQT and VDWQT allows updated pages to use a larger portion of the virtual buffer pool. Large DWQT and VDWQT can have a significant effect on the other thresholds. For example, in work load where pages are frequently updated, and the updated pages exceed the size of the virtual buffer pool, setting both DWQT and VDWQT to 90% would probably cause frequent threshold-reached events for sequential prefetch (and possibly the data management and immediate write).
Buffer Pool Write

Field Name: QBSTDWT
This is an exception field.

VERTI.DEF.WRITE THRESHOLD
The number of times the vertical deferred write threshold (VDWQT) was reached. This threshold is similar to the deferred write threshold but it applies to the number of updated pages for one single page set in the buffer pool. If the percentage or number of updated pages for the data set exceeds the threshold, writes up to 128 pages are scheduled for that data set.

Field Name: QBSTDWV
This is an exception field.

DM THRESHOLD
The number of times the data manager critical threshold (DMTH-95%) was reached.

This field shows how many times a page was immediately released because the data management threshold was reached.

The threshold is checked before a page is read or updated. If the threshold has not been exceeded, DB2 accesses the page in the virtual buffer pool once for each page, no matter how many rows are retrieved or updated in that page. If the threshold has been exceeded, Getpage requests and RELEASEs apply to rows instead of pages. That is, if more than one row is retrieved or updated in a page, more than one Getpage request and RELEASE is performed on that page.

Background and Tuning Information
Avoid reaching this threshold wherever possible because it significantly affects CPU usage. Set virtual buffer pool sizes large enough or reduce the workload on the buffer pool.

Field Name: QBSTDMC
This is an exception field.

PAGE-INS REQUIRED FOR WRITE
The number of page-ins required for a write I/O.

Field Name: QBSTWPI
This topic shows detailed information about “Statistics - Common Storage Below and Above 2 GB (DB2 10 or later)”.

### Statistics - Common Storage Below and Above 2 GB (DB2 10 or later)

The field labels shown in the following sample layout of “Statistics - Common Storage Below and Above 2 GB (DB2 10 or later)” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXTENDED CSA SIZE</td>
<td>The size of the common storage area (CSA) above the 16 MB line.</td>
</tr>
<tr>
<td>FIXED POOL BELOW</td>
<td>The amount of storage allocated for 31-bit common fixed pool storage.</td>
</tr>
<tr>
<td>VARIABLE POOL BELOW</td>
<td>The amount of storage allocated for 31-bit common variable pool storage.</td>
</tr>
<tr>
<td>GETMAIN BELOW</td>
<td>The amount of storage allocated for 31-bit common getmain storage.</td>
</tr>
<tr>
<td>FIXED POOL ABOVE</td>
<td>The amount of storage allocated for 64-bit common fixed pool storage.</td>
</tr>
<tr>
<td>VARIABLE POOL ABOVE</td>
<td>The amount of storage allocated for 64-bit common variable pool storage.</td>
</tr>
<tr>
<td>GETMAIN ABOVE</td>
<td>The amount of storage allocated for 64-bit common getmain storage.</td>
</tr>
<tr>
<td>STORAGE MANAGER CONTROL BLOCKS ABOVE</td>
<td>The amount of storage allocated for storage manager control blocks above.</td>
</tr>
<tr>
<td>REAL LOG MANAGER WRITE BUFFERS ABOVE</td>
<td>The amount of storage allocated for real log manager write buffers above.</td>
</tr>
<tr>
<td>REAL LOG MANAGER CONTROL BLOCKS ABOVE</td>
<td>The amount of storage allocated for real log manager control blocks above.</td>
</tr>
<tr>
<td>AUX LOG MANAGER CONTROL BLOCKS ABOVE</td>
<td>The amount of storage allocated for aux log manager control blocks above.</td>
</tr>
<tr>
<td>REAL STORAGE IN USE</td>
<td>The amount of storage in use.</td>
</tr>
<tr>
<td>AVERAGE THREAD FOOTPRINT</td>
<td>The average thread footprint.</td>
</tr>
<tr>
<td>AUXILIARY STORAGE IN USE</td>
<td>The amount of auxiliary storage in use.</td>
</tr>
</tbody>
</table>

### Field Details

- **EXTENDED CSA SIZE (MB)**
  - The size of the common storage area (CSA) above the 16 MB line.
  - **Field Name**: QW0225EC

- **FIXED POOL BELOW (MB)**
  - The amount of storage allocated for 31-bit common fixed pool storage.
  - **Field Name**: QW0225FC

- **VARIABLE POOL BELOW (MB)**
  - The amount of storage allocated for 31-bit common variable pool storage.
  - **Field Name**: QW0225VC

- **GETMAIN BELOW (MB)**
  - The amount of storage allocated for 31-bit common getmain storage.
  - **Field Name**: QW0225GC

- **FIXED POOL ABOVE (MB)**
  - The amount of storage allocated for 64-bit common fixed pool storage.
  - **Field Name**: QW0225FCG

- **VARIABLE POOL ABOVE (MB)**
  - The amount of storage allocated for 64-bit common variable pool storage.
  - **Field Name**: QW0225VCG
Common Storage Below and Above 2 GB

**GETMAINED ABOVE (MB)**

The amount of storage allocated for 64-bit common getmained storage.

*Field Name:* QW0225GCG

**STORAGE MANAGER CONTROL BLOCKS ABOVE (MB)**

The amount of storage allocated for 64-bit common storage for storage manager control structures.

*Field Name:* QW0225SMC

**REAL LOG MANAGER WRITE BUFFERS ABOVE (MB)**

The amount of real storage in the 64-bit common area in use for Log Manager write buffers.

*Field Name:* S225LWR

**REAL LOG MANAGER CONTROL BLOCKS ABOVE (MB)**

The amount of real storage in the 64-bit common area in use for Log Manager control blocks.

*Field Name:* S225LCR

**AUX LOG MANAGER CONTROL BLOCKS ABOVE (MB)**

The amount of auxiliary storage in the 64-bit common area in use for Log Manager control blocks.

*Field Name:* S225LAC

**REAL STORAGE IN USE (MB)**

The amount of real storage in use for 64-bit common storage. This is recorded at the subsystem level.

*Note:* This field is available in z/OS 1.10 (and maintenance) or later.

*Field Name:* S225CSR

**AVERAGE THREAD FOOTPRINT (MB)**

The current average real storage in use for common storage of active user threads (allied threads + active and pooled DBATs).

*Field Name:* S225CTFR

**AUXILIARY STORAGE IN USE (MB)**

The amount of auxiliary storage in use for 64-bit common storage. This is recorded at the subsystem level.

*Note:* This field is available in z/OS 1.10 (and maintenance) or later.

*Field Name:* S225CSA
CPU and Storage Metrics

This topic shows detailed information about “Statistics - CPU and Storage Metrics”.

This block shows information about CPU usage and storage metrics gathered by the z/OS Resource Measurement Facility (RMF) interface. The subsystem parameter ZOSMETRICS must be set to YES for enabling DB2 to retrieve data from RMF.

Statistics - CPU and Storage Metrics

The field labels shown in the following sample layout of “Statistics - CPU and Storage Metrics” are described in the following section.

<table>
<thead>
<tr>
<th>CPU AND STORAGE METRICS</th>
<th>QUANTITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>CP LPAR</td>
<td>4.00</td>
</tr>
<tr>
<td>CPU UTILIZATION LPAR</td>
<td>255.47</td>
</tr>
<tr>
<td>CPU UTILIZATION DB2</td>
<td>0.04</td>
</tr>
<tr>
<td>CPU UTILIZATION DB2 MSTR</td>
<td>0.00</td>
</tr>
<tr>
<td>CPU UTILIZATION DB2 DBM1</td>
<td>0.00</td>
</tr>
<tr>
<td>UNREFERENCED INTERVAL COUNT</td>
<td>65535.00</td>
</tr>
<tr>
<td>REAL STORAGE LPAR (MB)</td>
<td>3071.00</td>
</tr>
<tr>
<td>FREE REAL STORAGE LPAR (MB)</td>
<td>268.00</td>
</tr>
<tr>
<td>USED REAL STORAGE DB2 (MB)</td>
<td>240.00</td>
</tr>
<tr>
<td>VIRTUAL STORAGE LPAR (MB)</td>
<td>17051.26</td>
</tr>
<tr>
<td>FREE VIRTUAL STOR LPAR (MB)</td>
<td>13828.00</td>
</tr>
<tr>
<td>USED VIRTUAL STOR DB2 (MB)</td>
<td>332.00</td>
</tr>
</tbody>
</table>

CP LPAR

The number of central processors (CPs) on the logical partition (LPAR).

Field Name: QWOSLNCP

CPU UTILIZATION LPAR

The CPU utilization on the LPAR.

Field Name: QWOSLPRU

CPU UTILIZATION DB2

The CPU utilization of the DB2 subsystem.

Field Name: QWOSDB2U

CPU UTILIZATION DB2 MSTR

The CPU utilization of the DB2 MSTR address space.

Field Name: QWOSMSTU

CPU UTILIZATION DB2 DBM1

The CPU utilization of the DB2 DBM1 address space.

Field Name: QWOSDBMU

UNREFERENCED INTERVAL COUNT

The number of unreferenced intervals.

Field Name: QWOSLUIC

REAL STORAGE LPAR (MB)
CPU and Storage Metrics

The amount of real storage available on the LPAR.
Field Name: QWOSLRST

FREE REAL STORAGE LPAR (MB)
The amount of free real storage on the LPAR.
Field Name: QWOSLRSF

USED REAL STORAGE DB2 (MB)
The amount of real storage used for the DB2 subsystem.
Field Name: QWOSDRSU

VIRTUAL STORAGE LPAR (MB)
The amount of virtual storage available on the LPAR.
Field Name: QWOSLVST

FREE VIRTUAL STOR LPAR (MB)
The amount of free virtual storage on the LPAR.
Field Name: QWOSLVSF

USED VIRTUAL STOR DB2 (MB)
The amount of virtual storage used for the DB2 subsystem.
Field Name: QWOSDVSU
CPU Times

This topic shows detailed information about “Statistics - CPU Times”.

This block shows statistics data of CPU timer values for each resource manager and control address space.

DB2 can generate parallel tasks for the efficient execution of queries. Parallel tasks are executable units composed of special SRBs (service request block), which are called preemtatable SRBs. With preemtatable SRBs, the z/OS dispatcher can interrupt a task at any time to run other work at the same or higher dispatching priority. For non-distributed parallel work, parallel tasks run under a type of preemtatable SRB, which is called a client SRB. The client SRB lets the parallel task inherit the importance of the originating address space. For distributed requests, the parallel tasks run under a preemtatable SRB, which is called an enclave SRB.

Statistics - CPU Times

The field labels shown in the following sample layout of “Statistics - CPU Times” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSTCBT</td>
<td>TCB time for the system services address space.</td>
</tr>
<tr>
<td>SDTCBT</td>
<td>TCB time used for database services address space.</td>
</tr>
<tr>
<td>SDITCBT</td>
<td>IRLM TCB time.</td>
</tr>
<tr>
<td>SDDFTCBT</td>
<td>DDF address space TCB time.</td>
</tr>
<tr>
<td>SDLTLCBT</td>
<td>Total TCB time for all address spaces.</td>
</tr>
<tr>
<td>SSRSRB</td>
<td>The preemtatable SRB time for the system services address space, not including preemtatable SRB time consumed on an IBM zIIP.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSTCBT</td>
<td>0.120789 0.037248 0.027924 0.185961 0.000000 0.005635</td>
</tr>
<tr>
<td>SDTCBT</td>
<td>0.403656 0.218449 0.009574 0.631680 0.000000 0.019142</td>
</tr>
<tr>
<td>SDITCBT</td>
<td>0.000043 0.000000 0.224940 0.224983 0.000000 0.006818</td>
</tr>
<tr>
<td>SDDFTCBT</td>
<td>0.005795 1.122965 0.007127 1.135886 0.000000 0.034421</td>
</tr>
<tr>
<td>SDLTLCBT</td>
<td>0.530282 1.378663 0.269566 2.178511 0.000000 0.066015</td>
</tr>
</tbody>
</table>

Chapter 49. Statistics Report and Trace Blocks 49-41
**CPU Times**

The preemptable SRB time for the database services address space, not including preemptable SRB time consumed on an IBM zIIP.

*Field Name:* SDPSRB

**IRLM - PREEMPT SRB**

The preemptable SRB time for the IRLM address space, not including preemptable SRB time consumed on an IBM zIIP.

*Field Name:* SDIPSRB

**DDF ADDRESS SPACE - PREEMPT SRB**

The preemptable SRB time for the DDF address space, not including preemptable SRB time consumed on an IBM zIIP.

*Field Name:* SDDFPSRB

**TOTAL - PREEMPT SRB**

Total preemptable SRB time for all address spaces, not including preemptable SRB time consumed on an IBM zIIP.

*Field Name:* SDTLPSRB

**SYSTEM SERVICES ADDRESS SPACE - NONPREEMPT SRB**

The nonpreemptable SRB time for the system services address space.

*Field Name:* SSNPSR

**DATABASE SERVICES ADDRESS SPACE - NONPREEMPT SRB**

The nonpreemptable SRB time for the database services address space.

*Field Name:* SDNPSR

**IRLM - NONPREEMPT SRB**

The nonpreemptable SRB time for the IRLM address space.

*Field Name:* SDINPSR

**DDF ADDRESS SPACE - NONPREEMPT SRB**

The nonpreemptable SRB time for the DDF address space.

*Field Name:* SDDFNPSR

**TOTAL - NONPREEMPT SRB**

Total nonpreemptable SRB time for all address spaces.

*Field Name:* SDTLNPSR

**SYSTEM SERVICES ADDRESS SPACE - CP CPU TIME**

System services address space total time.

*Field Name:* SSTOTT

**DATABASE SERVICES ADDRESS SPACE - CP CPU TIME**

Database services address space total time.

*Field Name:* SDTOTT

**IRLM - CP CPU TIME**

IRLM address space total time.

*Field Name:* SDITOTT
DDF ADDRESS SPACE - CP CPU TIME
DDF address space total time.
Field Name: SDDFTOTT

TOTAL - CP CPU TIME
Total CPU time for all address spaces.
Field Name: SDTLTOTT

SYSTEM SERVICES ADDRESS SPACE - PREEMPT IIP SRB
The preemptable SRB time for the system services address space consumed on an IBM zIIP.
Field Name: SSPSRZ

DATABASE SERVICES ADDRESS SPACE - PREEMPT IIP SRB
The preemptable SRB time for the database services address space consumed on an IBM zIIP.
Field Name: SDPSRZ

IRLM - PREEMPT IIP SRB
The preemptable SRB time for the IRLM address space consumed on an IBM zIIP.
Field Name: SDIPSRZ

DDF ADDRESS SPACE - PREEMPT IIP SRB
The preemptable SRB time for the DDF address space consumed on an IBM zIIP.
Field Name: SDDFPSRZ

TOTAL - PREEMPT IIP SRB
Total preemptable SRB time for all address spaces consumed on an IBM zIIP.
Field Name: SDTLPSRZ
Data Capture

This topic shows detailed information about “Statistics - Data Capture”.

Statistics - Data Capture

The field labels shown in the following sample layout of “Statistics - Data Capture” are described in the following section.

<table>
<thead>
<tr>
<th>DATA CAPTURE</th>
<th>QUANTITY</th>
<th>/SECOND</th>
<th>/THREAD</th>
<th>/COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOG RECORDS CAPTURED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>LOG READS PERFORMED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>LOG RECORDS RETURNED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DATA ROWS RETURNED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DESCRIBES PERFORMED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DATA DESCRIPTIONS RETURNED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>TABLES RETURNED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
</tbody>
</table>

LOG RECORDS CAPTURED
The number of log records retrieved for which data capture processing was invoked.

Field Name: QWSDCDLC

LOG READS PERFORMED
The total number of data capture log reads for processing IFI reads requests for IFCID 185.

Field Name: QWSDCDLR

LOG RECORDS RETURNED
The total number of data capture log records returned.

Field Name: QWSDCDRR

DATA ROWS RETURNED
The total number of data capture data rows returned.

Field Name: QWSDCDDR

DESCRIBES PERFORMED
The total number of data capture describes performed.

A data capture describe is the process of getting descriptive information about a DB2 table from the catalog.

Field Name: QWSDCDMB

DATA DESCRIPTIONS RETURNED
The total number of data capture describes performed.

A data capture describe is the process of getting descriptive information about a DB2 table from the catalog.

Field Name: QWSDCDDD

TABLES RETURNED
The total number of data capture tables returned to the caller of the IFI reads call for IFCID 185.

Field Name: QWSDCDTB
Data Set Statistics

This topic shows detailed information about “Statistics - Data Set Statistics”.

Statistics - Data Set Statistics

The field labels shown in the following sample layout of “Statistics - Data Set Statistics” are described in the following section.

<table>
<thead>
<tr>
<th>BPOOL</th>
<th>DATABASE</th>
<th>TYPE</th>
<th>SYNCH I/O AVG</th>
<th>SYM I/O AVG DELAY</th>
<th>ASYM I/O AVG DELAY</th>
<th>CURRENT PAGES (VP)</th>
<th>CHANGED PAGES (VP)</th>
</tr>
</thead>
<tbody>
<tr>
<td>BP1</td>
<td>DBTPCCB</td>
<td>TSP</td>
<td>8.55</td>
<td>55.00</td>
<td>24.00</td>
<td>44.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TITEM000</td>
<td>N</td>
<td>0.02</td>
<td>8961.00</td>
<td>76.00</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>1</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BP1</td>
<td>DBTPCCB</td>
<td>IDX</td>
<td>7.02</td>
<td>51.00</td>
<td>12.00</td>
<td>46.00</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>XITEM000</td>
<td>N</td>
<td>0.00</td>
<td>5636.00</td>
<td>11.00</td>
<td>0.00</td>
<td>N/A</td>
</tr>
</tbody>
</table>

BPOOL

The name of the buffer pool to which this information refers.

Field Name: S199BPNM

DATABASE

Database name.

Field Name: S199DBNM

SPACENAM

Pageset name, which can be a table space or an index space.

This is derived from the internal pageset identifier. For a table space this is the value in the PSID column in SYSIBM.SYSTABLESPACE of the catalog when the DB2 trace record was written. For an index space, this is the value in the ISOBID column in SYSIBM.SYSINDEXES.

When OMEGAMON XE for DB2 PE cannot determine the pageset name, the ID is shown in hexadecimal.

Field Name: S199OBNM

PART

For a partitioned table space or index space, this is the partition number. For a nonpartitioned table space or index space, this is the data set number.

Field Name: QW0199DN

TYPE

This field indicates whether the pageset is a table space (T or TSP) or an index space (I or IDX).

Field Name: S199TYP

GBP

The value in this field specifies whether the pageset is group buffer pool dependent. This is only possible if DB2 has been set up for data sharing.

Field Name: S199GBP

SYNCH I/O AVG

Chapter 49. Statistics Report and Trace Blocks 49-45
**Data Set Statistics**

Average number of synchronous I/Os for the pageset, per second.
**Field Name:** S199SPAV

**ASYNC I/O AVG**
Average number of asynchronous I/Os for the pageset, per second.
**Field Name:** S199ACAV

**ASY I/O PGS AVG**
Average number of pageset pages read or written per asynchronous I/O.
**Field Name:** S199APAV

**SYN I/O AVG DELAY**
Average synchronous I/O delay for pages in the pageset, in milliseconds.
**Field Name:** QW0199SV

**SYN I/O MAX DELAY**
Maximum synchronous I/O delay for pages in the pageset, in milliseconds.
**Field Name:** QW0199SX

**ASYN I/O AVG DELAY**
Average asynchronous I/O delay for pages in the pageset, in milliseconds.
**Field Name:** QW0199AD

**ASYN I/O MAX DELAY**
The maximum asynchronous I/O delay for pages in the pageset, in milliseconds.
**Field Name:** QW0199AX

**CURRENT PAGES (VP)**
Number of pageset pages in the virtual buffer pool.
**Field Name:** QW0199VP

**CHANGED PAGES (VP)**
Number of changed page set pages in the virtual buffer pool.
**Field Name:** QW0199VD

**NUMBER OF GETPAGES**
The current number of Getpage requests.
**Field Name:** QW0199GP
This topic shows detailed information about “Statistics - Data Sharing Locking”.

In this example, the quantities per thread show as not calculated (N/C) because DB2 threads remained open during the reporting period.

### Statistics - Data Sharing Locking

The field labels shown in the following sample layout of “Statistics - Data Sharing Locking” are described in the following section.

<table>
<thead>
<tr>
<th>DATA SHARING LOCKING</th>
<th>QUANTITY</th>
<th>/SECOND</th>
<th>/THREAD</th>
<th>/COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLOBAL CONTENTION RATE (%)</td>
<td>0.68</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FALSE CONTENTION RATE (%)</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P/L-LOCKS XES RATE (%)</td>
<td>97.56</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LOCK REQUESTS (P-LOCKS)</td>
<td>56.00</td>
<td>0.02</td>
<td>N/C</td>
<td>N/C</td>
</tr>
<tr>
<td>UNLOCK REQUESTS (P-LOCKS)</td>
<td>34.00</td>
<td>0.01</td>
<td>N/C</td>
<td>N/C</td>
</tr>
<tr>
<td>CHANGE REQUESTS (P-LOCKS)</td>
<td>22.00</td>
<td>0.01</td>
<td>N/C</td>
<td>N/C</td>
</tr>
<tr>
<td>SYNCH.XES - LOCK REQUESTS</td>
<td>3759.00</td>
<td>1.06</td>
<td>N/C</td>
<td>N/C</td>
</tr>
<tr>
<td>SYNCH.XES - CHANGE REQUESTS</td>
<td>7.00</td>
<td>0.00</td>
<td>N/C</td>
<td>N/C</td>
</tr>
<tr>
<td>SYNCH.XES - UNLOCK REQUESTS</td>
<td>3770.00</td>
<td>1.07</td>
<td>N/C</td>
<td>N/C</td>
</tr>
<tr>
<td>BACKGROUND.XES - CHILD LOCKS</td>
<td>4.00</td>
<td>0.00</td>
<td>N/C</td>
<td>N/C</td>
</tr>
<tr>
<td>ASYNCH.XES - CONVERTED LOCKS</td>
<td>13.00</td>
<td>0.00</td>
<td>N/C</td>
<td>N/C</td>
</tr>
<tr>
<td>SUSPENDS - IRLM GLOBAL CONT</td>
<td>52.00</td>
<td>0.01</td>
<td>N/C</td>
<td>N/C</td>
</tr>
<tr>
<td>SUSPENDS - XES GLOBAL CONT</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>N/C</td>
</tr>
<tr>
<td>SUSPENDS - FALSE CONT. MBR</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>N/C</td>
</tr>
<tr>
<td>SUSPENDS - FALSE CONT. LPAR</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>NO DELAY LOCK REQ REJECTS</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>N/C</td>
</tr>
<tr>
<td>INCOMPATIBLE RETAINED LOCK</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>N/C</td>
</tr>
<tr>
<td>NOTIFY MESSAGES SENT</td>
<td>28.00</td>
<td>0.01</td>
<td>N/C</td>
<td>N/C</td>
</tr>
<tr>
<td>NOTIFY MESSAGES RECEIVED</td>
<td>42.00</td>
<td>0.01</td>
<td>N/C</td>
<td>N/C</td>
</tr>
<tr>
<td>P-LOCK/NOTIFY EXITS ENGINES</td>
<td>500.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>P-LCK/NFY EX.ENGINE UNAVAL</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>N/C</td>
</tr>
<tr>
<td>PSET/PART P-LCK NEGOTIATION</td>
<td>20.00</td>
<td>0.01</td>
<td>N/C</td>
<td>N/C</td>
</tr>
<tr>
<td>PAGE P-LOCK NEGOTIATION</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>N/C</td>
</tr>
<tr>
<td>OTHER P-LOCK NEGOTIATION</td>
<td>12.00</td>
<td>0.00</td>
<td>N/C</td>
<td>N/C</td>
</tr>
<tr>
<td>P-LOCK CHANGE DURING NEG.</td>
<td>30.00</td>
<td>0.01</td>
<td>N/C</td>
<td>N/C</td>
</tr>
</tbody>
</table>

**GLOBAL CONTENTION RATE (%)**

The total number of suspends because of contention divided by the total number of synchronous requests that went to XES, and the lock requests that were converted from synchronous to asynchronous locks, and the locks because of child lock propagation.

**Note:** If multiple members from the same data sharing group run on the same LPAR, the global contention rate should be ignored for a member where the QTGSFCON flag is zero. The QTGSFCON flag indicates whether the false contention is reported at the subsystem (=1) or LPAR level (=0).

**Field Name:** SGLOBRAT

**FALSE CONTENTION RATE (%)**

The total number of suspends because of false contention divided by the total number of synchronous requests that went to XES, and the lock requests that were converted from synchronous to asynchronous locks, and
the locks because of child lock propagation. A false contention is if two different locks on different resources hash to the same lock entry.

Note: If multiple members from the same data sharing group run on the same LPAR, the global contention rate should be ignored for a member where the QTGSFCON flag is zero. The QTGSFCON flag indicates whether the false contention is reported at the subsystem (=1) or LPAR level (=0).

Background and Tuning Information
Try to keep the false contention rate to no more than 50% of the total global lock contention.

Field Name: SFLSERAT
P/L-LOCKS XES RATE (%)
Shows the percentage of P/L-lock requests that were propagated to XES synchronously.

Background and Tuning Information
This number reflects the effects of explicit hierarchical locking and other locking optimizations. Assuming a 100% Data Sharing workload, a value of 94% would mean that 6% of all Transaction Locks were not propagated to XES due to Data Sharing locking optimizations.

DB2 has some optimizations to reduce the necessity to go beyond the local IRLM whenever possible:
• Explicit hierarchical locking allows IRLM to grant child locks locally when there is no inter-DB2 R/W interest on the parent.
• If there is a single DB2 with update interest and multiple DB2s with read-only interest, DB2 propagates fewer locks than when all DB2s have update interest in the page set.
• All locks that go beyond the local IRLM are owned by the subsystem, not by an individual work unit. This allows for another optimization. Only the most restrictive lock mode for an object on a given subsystem must be propagated to XES and the coupling facility. A new lock that is equal to or less restrictive than one currently being held is not propagated.

Field Name: SLLOCRA
LOCK REQUESTS (P-LOCKS)
The number of lock requests for P-locks.

Field Name: QTGSLPLK
UNLOCK REQUESTS (P-LOCKS)
The number of unlock requests for P-locks.

Field Name: QTGSUPLK
CHANGE REQUESTS (P-LOCKS)
The number of change requests for P-locks.

Field Name: QTGSCPLK
SYNCH.XES - LOCK REQUESTS
The number of P/L-lock requests propagated to z/OS XES synchronously.
**Data Sharing Locking**

This number is not incremented if the request is suspended before going to XES.

**Field Name**: QTGSLSLM

**SYNCH.XES - CHANGE REQUESTS**

The number of change requests propagated to z/OS XES synchronously, including logical and physical locks.

This number is not incremented if the request is suspended before going to XES.

**Field Name**: QTGSCSLM

**SYNCH.XES - UNLOCK REQUESTS**

The number of unlock requests propagated to z/OS XES synchronously, including logical and physical locks.

This number is not incremented if the request is suspended before going to XES.

**Field Name**: QTGSUSLM

**BACKGROUND.XES - CHILD LOCKS**

The number of resources propagated by IRLM to z/OS XES asynchronously, including logical and physical locks.

This can happen when new inter-DB2 interest occurs on a parent resource or when a request completes after the requester's execution unit was suspended.

**Field Name**: QTGSKIDS

**ASYNCH.XES - CONVERTED LOCKS**

The number of synchronous to asynchronous heuristic conversions for LOCK requests in XES. This conversion is done when XES determines that it is more efficient to drive the request asynchronously to the coupling facility (CF).

**Field Name**: QTGSFLSE

**SUSPENDS - IRLM GLOBAL CONT**

The number of suspensions due to IRLM global resource contention. All IRLM lock states were in conflict on the same resource.

Global contention requires intersystem communication to resolve the lock conflict whereas local contention does not.

**Field Name**: QTGSIGLO

**SUSPENDS - XES GLOBAL CONT**

The number of suspensions due to z/OS XES global resource contention. The z/OS XES lock states were in conflict but the IRLM lock states were not.

IRLM has many lock states but XES is only aware of the exclusive and shared lock states.

**Field Name**: QTGSSGLO

**SUSPENDS - FALSE CONT. MBR**
Data Sharing Locking

The total number of false contentions for LOCK and UNLOCK requests. A false contention occurs when different resource names hash to the same entry in the coupling facility (CF) lock table. The CF detects contention within the hash entry, and XES uses intersystem messaging to determine that no actual resource contention exists.

**Note:** The QTGSFCON flag indicates whether the false contention is reported at subsystem (=1) or LPAR level (=0).

**Field Name:** STGSFLM1

**SUSPENDS - FALSE CONT. LPAR**

The total number of false contentions for LOCK and UNLOCK requests. A false contention occurs when different resource names hash to the same entry in the coupling facility (CF) lock table. The CF detects contention within the hash entry, and XES uses intersystem messaging to determine that no actual resource contention exists.

**Note:** The QTGSFCON flag indicates whether the false contention is reported at subsystem (=1) or LPAR level (=0).

**Field Name:** STGSFLM2

**NO DELAY LOCK REQ REJECTS**

The total number of failed DB2 lock requests to XES to process without delay. XES rejects the lock request because it could not process it synchronously.

**Field Name:** QTGSCREJ

**INCOMPATIBLE RETAINED LOCK**

The number of global lock or change requests denied or suspended due to an incompatible retained lock.

**Field Name:** QTGSDRTA

**NOTIFY MESSAGES SENT**

The number of notify messages sent.

**Field Name:** QTGSNNTFY

**NOTIFY MESSAGES RECEIVED**

The number of notify messages received.

**Field Name:** QTGSNTFR

**P-LOCK/NOTIFY EXITS ENGINES**

The maximum number of engines available for physical lock exit or notify exit requests.

**Field Name:** QTGSPEMX

**P-LCK/NFY EX.ENGINE UNAVAIL**

The number of times an engine is not available for physical lock exit or notify exit requests.

**Field Name:** QTGSPEQW

**PSET/PART P-LCK NEGOTIATION**
Data Sharing Locking

The number of times this DB2 was driven to negotiate a partition or page set physical lock due to changing inter-DB2 interest levels on the partition or page set.

Field Name: QTGSPPPE

PAGE P-LOCK NEGOTIATION

The number of times this DB2 negotiated a page physical lock because of physical lock contention within DB2.

Field Name: QTGSPGPE

OTHER P-LOCK NEGOTIATION

The number of times this DB2 was driven to negotiate a physical lock type other than page set, partition, or page.

Field Name: QTGSOTPE

P-LOCK CHANGE DURING NEG.

The number of times a physical lock change request was issued during physical lock negotiation.

Field Name: QTGSCHNP
This topic shows detailed information about “Statistics - DBM1 and MVS Storage Below 2 GB”.

This block shows information about storage allocation within the DBM1 address space.

Storage quantities are shown in megabytes, this means that when you want to compare this with absolute values, as stored in the performance database, for example, you need to multiply the value shown by 1048576 (1024*1024). Similarly where a quantity is shown followed by a K, for example 262.1K, the quantity shown is 262.1MB*1000 (262.1*1048576*1000 bytes).

Statistics - DBM1 and MVS Storage Below 2 GB

The field labels shown in the following sample layout of “Statistics - DBM1 and MVS Storage Below 2 GB” are described in the following section.

<table>
<thead>
<tr>
<th>DBM1 AND MVS STORAGE BELOW 2 GB</th>
<th>QUANTITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>TOTAL DBM1 STORAGE BELOW 2 GB</td>
<td>6.52</td>
</tr>
<tr>
<td>TOTAL GETMAINED STORAGE</td>
<td>0.52</td>
</tr>
<tr>
<td>EDM POOL</td>
<td>0.00</td>
</tr>
<tr>
<td>TOTAL VARIABLE STORAGE</td>
<td>1.11</td>
</tr>
<tr>
<td>TOTAL AGENT LOCAL STORAGE</td>
<td>0.32</td>
</tr>
<tr>
<td>TOTAL AGENT SYSTEM STORAGE</td>
<td>0.22</td>
</tr>
<tr>
<td>NUMBER OF PREFETCH ENGINES</td>
<td>9.00</td>
</tr>
<tr>
<td>NUMBER OF DEFERRED WRITE ENGINES</td>
<td>1.00</td>
</tr>
<tr>
<td>NUMBER OF CASTOUT ENGINES</td>
<td>7.00</td>
</tr>
<tr>
<td>NUMBER OF GBP WRITE ENGINES</td>
<td>1.00</td>
</tr>
<tr>
<td>NUMBER OF P-LOCK/NOTIFY EXIT ENGINES</td>
<td>2.00</td>
</tr>
<tr>
<td>TOTAL AGENT NON-SYSTEM STORAGE</td>
<td>0.10</td>
</tr>
<tr>
<td>TOTAL NUMBER OF ACTIVE USER THREADS</td>
<td>7.00</td>
</tr>
<tr>
<td>NUMBER OF ALLIED THREADS</td>
<td>7.00</td>
</tr>
<tr>
<td>NUMBER OF ACTIVE DBATS</td>
<td>0.00</td>
</tr>
<tr>
<td>NUMBER OF POOLED DBATS</td>
<td>0.00</td>
</tr>
<tr>
<td>NUMBER OF PARALLEL CHILD THREADS</td>
<td>0.00</td>
</tr>
<tr>
<td>RID POOL</td>
<td>N/A</td>
</tr>
<tr>
<td>PIPE MANAGER SUB POOL</td>
<td>N/A</td>
</tr>
<tr>
<td>LOCAL DYNAMIC STMT CACHE_CNTL_BLK</td>
<td>N/A</td>
</tr>
<tr>
<td>SYSTEM COPIES OF CACHED SQL_STMTS</td>
<td>N/A</td>
</tr>
<tr>
<td>IN USE STORAGE</td>
<td>N/A</td>
</tr>
<tr>
<td>STATEMENTS COUNT</td>
<td>N/A</td>
</tr>
<tr>
<td>HWM FOR ALLOCATED STATEMENTS</td>
<td>N/A</td>
</tr>
<tr>
<td>STATEMENT COUNT AT HWM</td>
<td>N/A</td>
</tr>
<tr>
<td>DATE AT HWM</td>
<td>N/A</td>
</tr>
<tr>
<td>TIME AT HWM</td>
<td>N/A</td>
</tr>
<tr>
<td>SYSTEM COPIES OF STATIC_SQL</td>
<td>N/A</td>
</tr>
<tr>
<td>IN USE STORAGE</td>
<td>N/A</td>
</tr>
<tr>
<td>THREAD PLAN AND PACKAGE STORAGE</td>
<td>N/A</td>
</tr>
<tr>
<td>BUFFER MANAGER STORAGE_CNTL_BLK</td>
<td>0.35</td>
</tr>
<tr>
<td>TOTAL FIXED STORAGE</td>
<td>0.08</td>
</tr>
<tr>
<td>TOTAL GETMAINED STACK STORAGE</td>
<td>4.80</td>
</tr>
<tr>
<td>TOTAL STACK STORAGE IN USE</td>
<td>4.57</td>
</tr>
<tr>
<td>SYSTEM AGENT STACK STORAGE IN USE</td>
<td>3.94</td>
</tr>
<tr>
<td>STORAGE CUSHION</td>
<td>329.52</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DBM1 AND MVS STORAGE BELOW 2 GB CONTINUED</th>
<th>QUANTITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>24 BIT LOW PRIVATE</td>
<td>0.22</td>
</tr>
<tr>
<td>24 BIT HIGH PRIVATE</td>
<td>0.45</td>
</tr>
<tr>
<td>24 BIT PRIVATE CURRENT HIGH ADDRESS</td>
<td>00000000000003E000</td>
</tr>
</tbody>
</table>
31 BIT EXTENDED LOW PRIVATE (MB) 74.39
31 BIT EXTENDED HIGH PRIVATE (MB) 27.69
31 BIT PRIVATE CURRENT HIGH ADDRESS 000000026F0D000
EXTENDED REGION SIZE (MAX) (MB) 1524.00
EXTENDED CSA SIZE (MB) 300.03
AVERAGE THREAD FOOTPRINT (MB) 0.25
MAX NUMBER OF POSSIBLE THREADS 3984
AVERAGE THREAD FOOTPRINT (TYPE II) (MB) 0.09
MAX NUMBER OF POSSIBLE THREADS (TYPE II) 13581

TOTAL DBM1 STORAGE BELOW 2 GB (MB)
Total DBM1 storage. This includes:
- Fixed length storage use
- Getmained storage
- Save areas
- Variables

Field Name: SW0225DB

TOTAL GETMAINED STORAGE (MB)
Total storage acquired by GETMAIN. This includes space for virtual pools, EDM pool, compression dictionary, castout buffers, and the data space lookaside buffer, hiperpool control blocks, and data space buffer pool control blocks.

This figure can be different from the sum of GETMAIN storage items shown in the statistics DBM1 storage, because DB2 does not produce grouping statistics for all GETMAIN storage.

Field Name: QW0225GM

EDM POOL (MB)
Storage used for EDM pool.

Field Name: SISEPAGE

TOTAL VARIABLE STORAGE (MB)
Total storage used by all variable pools. This includes storage used by:
- System agents
- Local agents
- RID pool
- Pipe manager subpool
- Local dynamic statement cache control blocks
- Local dynamic statement cache statement pool
- Buffer and data manager trace tables
- A list of objects in restricted state including the new PRO state. If consumption of this storage pool is high, review restrictive exception state of database objects and check whether they can be resolved or reduced.

Field Name: QW0225VR

TOTAL AGENT LOCAL STORAGE (MB)
The amount of storage allocated for agent-related local storage. This storage is used for operations such as sort.
Background and Tuning Information

Sorting requires a large amount of virtual storage because there can be multiple copies of the data being sorted at a given time.

DB2 Sort uses two kinds of storage pool for various internal control structures and data records, an agent-related local storage pool and a global sort pool. To take advantage of the 64-bit addressability for larger storage pool, some high level sort control structures are kept in agent-related storage below the 2 GB bar, which contain 64-bit pointers to areas in the global sort pool above the 2 GB bar. The sort pool above 2 GB contains sort tree nodes and data buffers.

Field Name: QW0225AL

TOTAL AGENT SYSTEM STORAGE (MB)

Storage used by system agents.

Field Name: QW0225AS

NUMBER OF PREFETCH ENGINES

Number of engines used for sequential, list, and dynamic prefetch.

Field Name: QW0225PF

NUMBER OF DEFERRED WRITE ENGINES

Number of engines used for deferred write operations.

Field Name: QW0225DW

NUMBER OF CASTOUT ENGINES

Number of engines available for data-sharing castout processing.

Field Name: QW0225CE

NUMBER OF GBP WRITE ENGINES

Number of engines for group buffer pool writes.

Field Name: QW0225GW

NUMBER OF P-LOCK/NOTIFY EXIT ENGINES

Number of data sharing P-Lock engines and Notify Exit engines.

Field Name: QW0225PL

TOTAL AGENT NON-SYSTEM STORAGE (MB)

Total Agent Non-System Storage. It is the difference between the Total Agent Local Storage (QW0225AL) and the Total Agent System Storage (QW0225AS).

Field Name: SW0225AN

TOTAL NUMBER OF ACTIVE USER THREADS

Total number of active user threads. This includes all active allied threads and the current number of active DBATs.

Field Name: SACUSTHR

NUMBER OF ALLIED THREADS

The number of active allied threads.

Field Name: QW0225AT
NUMBER OF ACTIVE DBATS
The number of active connections, or active and disconnected DBAT
threads.
Field Name: SACDBATS

NUMBER OF POOLED DBATS
The current number of disconnected (pooled) DBATs that are available to
process type 2 inactive or new connections.
Field Name: QDSTNADS

NUMBER OF PARALLEL CHILD THREDS
The number of active parallel child threads.
Field Name: QW0225PT

RID POOL (MB)
Storage for RID list processing such as list prefetch, index ANDing, and
ORing.

Note:
• Prior to DB2 10, this value is shown in the “DBM1 AND MVS BELOW 2
  GB” block.
• For DB2 10 or later, this value is shown in the “DBM1 STORAGE
  ABOVE 2 GB” block.
Field Name: QW0225RP

PIPE MANAGER SUB POOL (MB)
Storage allocated to Pipe Manager for parallel query operations.
Field Name: QW0225PM

LOCAL DYNAMIC STMT CACHE CNTL BLKS (MB)
Storage for local dynamic statement cache control blocks.
Field Name: QW0225SB

SYSTEM COPIES OF CACHED SQL STMTS (MB)
The total shareable storage allocated for dynamic SQL statements used by
active threads.
• For DB2 11, this field is derived from QW0225SC8 and related to storage
  above the bar.
• Prior to DB2 11, this field is derived from QW0225SC and related to
  storage below the bar.
• For DB2 10, the storage is used for executable code sequences (xPROC).
• For DB2 9, it is the storage allocated for the local cache storage pool.
Field Name: SW0225SC

IN USE STORAGE (MB)
The total shareable storage requested for dynamic SQL statements used by
active threads.
• For DB2 11, this field is derived from QW0225LS8 and related to storage
  above the bar.
Prior to DB2 11, this field is derived from QW0225LS and related to storage below the bar.

For DB2 10, the storage is used for executable code sequences (xPROC).

For DB2 9, it is the storage used for thread copies in the local cache storage pool.

Field Name: SW0225LS

STATISTICS COUNT
The number of dynamic SQL local cache statements used by active threads.

For DB2 10 or later, this value is related to shared agent local variable pools above the bar.

Prior to DB2 10, this value is related to the local cache storage pool below the bar.

Field Name: QW0225LC

HWM FOR ALLOCATED STATEMENTS (MB)
A statistics interval high-water mark of requested shareable storage for dynamic SQL statements used by active threads.

For DB2 11, this field is derived from QW0225HS8 and related to storage above the bar.

Prior to DB2 11, this field is derived from QW0225HS and related to storage below the bar.

For DB2 10, the storage is used for executable code sequences (xPROC).

For DB2 9, it is the storage used for thread copies in the local cache storage pool.

Field Name: SW0225HS

STATEMENT COUNT AT HWM
The number of dynamic SQL local cache statements used by active threads at high storage time.

For DB2 10 or later, this value is related to shared agent local variable pools above the bar.

Prior to DB2 10, this value is related to the local cache storage pool below the bar.

Field Name: QW0225HC

DATE AT HWM
The timestamp at high-water storage.

Field Name: QW0225HT

TIME AT HWM
The timestamp at high-water storage.

Field Name: QW0225HT

SYSTEM COPIES OF STATIC SQL (MB)
The total shareable storage allocated for static SQL statements.

For DB2 11, this field is derived from QW0225SX8 and related to storage above the bar.

Prior to DB2 11, this field is derived from QW0225SX and used for storage of executable code sequences (xPROC) below the bar.
Field Name: SW0225SX

IN USE STORAGE (MB)

   The total storage requested for shareable static SQL statements.
   
   Prior to DB2 11, this field is derived from QISEKSPA8 and used for
   storage of executable code sequences (xPROC) below the bar.
   
   For DB2 11, this field is derived from QISEKSPA and related to storage
   above the bar.

Field Name: SISEKSPA

THREAD PLAN AND PACKAGE STORAGE (MB)

   The storage allocated to plans and packages below the bar.

Field Name: SISESQB

BUFFER MANAGER STORAGE CNTL BLKS (MB)

   Storage used for page set control blocks.

Field Name: QW0225BB

TOTAL FIXED STORAGE (MB)

   Total amount of fixed storage.

Field Name: QW0225FX

TOTAL GETMAINED STACK STORAGE (MB)

   Total GETMAINED storage allocated for program stack use.

Field Name: QW0225GS

TOTAL STACK STORAGE IN USE (MB)

   The amount of stack storage that is in use.

Field Name: QW0225SU

SYSTEM AGENT STACK STORAGE IN USE (MB)

   The amount of 31-bit stack storage that is in use for system agents. This is
   a subset of QW0225SU.

Field Name: QW0225SS

STORAGE CUSHION (MB)

   Storage reserved to allow DB2 to complete critical functions while short on
   storage. This includes the contract warning cushion, storage reserved for
   must-complete operations, and storage for MVS use.

Field Name: STORCUSH

24 BIT LOW PRIVATE (MB)

   The amount of private MVS storage below the 16 MB line. This storage is
   obtained from bottom upward, usually for unauthorized programs.

Field Name: QW0225LO

24 BIT HIGH PRIVATE (MB)

   The amount of private MVS storage below the 16 MB line. This storage is
   obtained from top downward, usually for authorized programs.

Field Name: QW0225HI
**DBM1 and MVS Storage Below 2 GB**

**24 BIT PRIVATE CURRENT HIGH ADDRESS**
The current high address of the 24-bit private region.

*Field Name:* QW0225TP

**31 BIT EXTENDED LOW PRIVATE (MB)**
The amount of private MVS storage above the 16 MB line. This storage is obtained from bottom upward, usually for unauthorized programs.

*Field Name:* QW0225EL

**31 BIT EXTENDED HIGH PRIVATE (MB)**
The amount of private MVS storage above the 16 MB line. This storage is obtained from top downward, usually for authorized programs.

*Field Name:* QW0225EH

**31 BIT PRIVATE CURRENT HIGH ADDRESS**
The current high address of the 31-bit private region.

*Field Name:* QW0225EP

**EXTENDED REGION SIZE (MAX) (MB)**
The maximum amount of MVS private storage available above the 16 MB line.

*Field Name:* QW0225RG

**EXTENDED CSA SIZE (MB)**
The size of the common storage area (CSA) above the 16 MB line.

*Field Name:* QW0225EC

**AVERAGE THREAD FOOTPRINT (MB)**
The current average memory usage of active user threads (allied threads and DBATs).

*Field Name:* SW0225TF

**MAX NUMBER OF POSSIBLE THREADS**
The maximum number of possible threads. It depends on the storage size and average memory usage of active user threads.

*Field Name:* SW0225MT

**AVERAGE THREAD FOOTPRINT (TYPE II) (MB)**
The current average memory usage of active allied threads and the maximum number of active DBATs that existed. The formula used for this value is suited for Enterprise Resource Planning (ERP) systems, such as SAP.

*Field Name:* SW0225TS

**MAX NUMBER OF POSSIBLE THREADS (TYPE II)**
The maximum number of possible threads. It depends on the storage size and average memory usage of active allied threads and the maximum number of active DBATs that existed.

*Field Name:* SW0225MS
This topic shows detailed information about “Statistics - DBM1 Storage Above 2 GB”.

Statistics - DBM1 Storage Above 2 GB

The field labels shown in the following sample layout of “Statistics - DBM1 Storage Above 2 GB” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>GETMAINED STORAGE (MB)</td>
<td>1505.40</td>
</tr>
<tr>
<td>FIXED STORAGE (MB)</td>
<td>10.87</td>
</tr>
<tr>
<td>VARIABLE STORAGE (MB)</td>
<td>235.22</td>
</tr>
<tr>
<td>COMPRESSION DICTIONARY (MB)</td>
<td>19.35</td>
</tr>
<tr>
<td>IN USE EDM DBD POOL (MB)</td>
<td>0.42</td>
</tr>
<tr>
<td>IN USE EDM STATEMENT POOL (MB)</td>
<td>1.49</td>
</tr>
<tr>
<td>IN USE EDM RDS POOL (MB)</td>
<td>N/A</td>
</tr>
<tr>
<td>IN USE EDM SKELETON POOL (MB)</td>
<td>0.11</td>
</tr>
<tr>
<td>STORAGE MANAGER CONTROL BLOCKS (MB)</td>
<td>7588.12</td>
</tr>
<tr>
<td>VIRTUAL BUFFER POOLS (MB)</td>
<td>270.40</td>
</tr>
<tr>
<td>VIRTUAL POOL CONTROL BLOCKS (MB)</td>
<td>0.00</td>
</tr>
<tr>
<td>CASTOUT BUFFERS (MB)</td>
<td>0.00</td>
</tr>
<tr>
<td>SHARED GETMAINED STORAGE (MB)</td>
<td>48.25</td>
</tr>
<tr>
<td>SHARED FIXED STORAGE (MB)</td>
<td>4.84</td>
</tr>
<tr>
<td>RID POOL (MB)</td>
<td>2348.71</td>
</tr>
<tr>
<td>SHARED VARIABLE STORAGE (MB)</td>
<td>2035.94</td>
</tr>
<tr>
<td>TOTAL AGENT LOCAL STORAGE (MB)</td>
<td>30.30</td>
</tr>
<tr>
<td>TOTAL AGENT SYSTEM STORAGE (MB)</td>
<td>2905.64</td>
</tr>
<tr>
<td>TOTAL AGENT NON-SYSTEM STORAGE (MB)</td>
<td>5.20</td>
</tr>
<tr>
<td>THREAD COPIES OF Cached SQL STMTS (MB)</td>
<td>24.33</td>
</tr>
<tr>
<td>IN USE STORAGE (MB)</td>
<td>46.05</td>
</tr>
<tr>
<td>STATEMENTS COUNT</td>
<td>50.00</td>
</tr>
<tr>
<td>HWM FOR ALLOCATED STATEMENTS (MB)</td>
<td>50.00</td>
</tr>
<tr>
<td>STATEMENT COUNT AT HWM</td>
<td>0.00</td>
</tr>
<tr>
<td>DATE AT HWM</td>
<td>07/05/11</td>
</tr>
<tr>
<td>TIME AT HWM</td>
<td>03:40:29.08</td>
</tr>
<tr>
<td>DYNAMIC STMT CACHE CNTL BLKS (MB)</td>
<td>121.52</td>
</tr>
<tr>
<td>SYSTEM COPIES OF Cached SQL STMTS (MB)</td>
<td>1536.00</td>
</tr>
<tr>
<td>IN USE STORAGE (MB)</td>
<td>483.87</td>
</tr>
<tr>
<td>HWM FOR ALLOCATED STATEMENTS (MB)</td>
<td>50.00</td>
</tr>
<tr>
<td>SYSTEM COPIES OF STATIC SQL (MB)</td>
<td>50.00</td>
</tr>
<tr>
<td>IN USE STORAGE (MB)</td>
<td>87.75</td>
</tr>
<tr>
<td>THREAD PLAN AND PACKAGE STORAGE (MB)</td>
<td>0.00</td>
</tr>
<tr>
<td>ARRAY VARIABLE STORAGE (MB)</td>
<td>05/11</td>
</tr>
<tr>
<td>SHARED STORAGE MANAGER CNTL BLKS (MB)</td>
<td>29.08</td>
</tr>
<tr>
<td>SHARED SYSTEM AGENT STACK STORAGE (MB)</td>
<td>21.52</td>
</tr>
<tr>
<td>STACK STORAGE IN USE (MB)</td>
<td>36.00</td>
</tr>
<tr>
<td>SHARED NON-SYSTEM AGENT STACK STORAGE (MB)</td>
<td>83.87</td>
</tr>
<tr>
<td>STACK STORAGE IN USE (MB)</td>
<td>0.00</td>
</tr>
</tbody>
</table>

GETMAINED STORAGE (MB)

Total storage acquired by GETMAIN. This includes space for the compression dictionary, and statement and DBD cache that can be used by the Environmental Descriptor Manager (EDM).

This figure can be different from the sum of GETMAIN storage items shown in the statistics DBM1 storage, because DB2 does not produce grouping statistics for all GETMAIN storage.

Field Name: QW0225GA

FIXED STORAGE (MB)
DBM1 Storage Above 2 GB

The total amount of fixed storage above the 2 GB bar.
Field Name: QW0225FA

VARIABLE STORAGE (MB)
Amount of variable storage available above the 2 GB bar.
Field Name: QW0225VA

COMPRESSION DICTIONARY (MB)
Storage space allocated for the compression dictionary.
Field Name: QW0225CD

IN USE EDM DBD POOL (MB)
The amount of storage used by database descriptors in the EDM DBD pool above the 2 GB bar.
Field Name: SISEDDBDP

IN USE EDM STATEMENT POOL (MB)
The amount of storage used by cached dynamic SQL statements in the EDM Statement pool above the 2 GB bar.
Field Name: SISEDYNP

IN USE EDM RDS POOL (MB)
The amount of storage used by objects in the EDM RDS pool above the 2 GB bar.
Field Name: SISECPTA

IN USE EDM SKELETON POOL (MB)
The amount of storage used by objects in the EDM Skeleton pool above the 2 GB bar.
Field Name: SISESKCP

STORAGE MANAGER CONTROL BLOCKS (MB)
Total 64-bit storage allocated for storage manager control structures.
Field Name: QW0225SM

VIRTUAL BUFFER POOLS (MB)
Total storage allocated for virtual buffer pools above the 2 GB bar.
Field Name: SVPOOLZ

VIRTUAL POOL CONTROL BLOCKS (MB)
Storage used for primary virtual pool control blocks above the 2 GB bar.
Field Name: SBSTVPLZ

CASTOUT BUFFERS (MB)
Storage used for castout buffers.
Field Name: SW0225C2

SHARED GETMAINED STORAGE (MB)
The amount of virtual shared storage acquired by GETMAIN above the 2 GB bar.
Field Name: QW0225SG

SHARED FIXED STORAGE (MB)

The amount of total fixed virtual shared storage above the 2 GB bar.

Field Name: QW0225SF

RID POOL (MB)

Storage for RID list processing such as list prefetch, index ANDing, and ORing.

Note:
- Prior to DB2 10, this value is shown in the “DBM1 AND MVS BELOW 2 GB” block.
- For DB2 10 or later, this value is shown in the “DBM1 STORAGE ABOVE 2 GB” block.

Field Name: QW0225RP

SHARED VARIABLE STORAGE (MB)

The amount of virtual shared variable storage above the 2 GB bar.

Field Name: QW0225SV

TOTAL AGENT LOCAL STORAGE (MB)

The amount of storage allocated for agent-related 64-bit local storage (DB2 field: QW0225ALG).

Field Name: SW225ALG

TOTAL AGENT SYSTEM STORAGE (MB)

The amount of 64-bit storage used by system agents (DB2 field: QW0225ASG).

Field Name: SW225ASG

TOTAL AGENT NON-SYSTEM STORAGE (MB)

The amount of 64-bit storage used by non-system agents. It is the difference between the Total Agent Local Storage (QW0225ALG) and the Total Agent System Storage (QW0225ASG).

Field Name: SW225ANG

THREAD COPIES OF CACHED SQL STMTS (MB)

This field is provided for consistency purposes. It has a value of N/A. The value can be estimated by the HWM FOR ALLOCATED STATEMENTS (QW0225H2).

Field Name: SW0225DY

IN USE STORAGE (MB)

The total non-shareable storage requested for dynamic SQL statements used by active threads.

Note: For DB2 10 or later, this value is related to shared agent local variable pools above the bar.

Field Name: QW0225L2

STATEMENTS COUNT
The number of dynamic SQL local cache statements used by active threads.

- For DB2 10 or later, this value is related to shared agent local variable pools above the bar.
- Prior to DB2 10, this value is related to the local cache storage pool below the bar.

**Field Name:** QW0225LC

**HWM FOR ALLOCATED STATEMENTS (MB)**

A statistics interval high-water mark of requested non-shareable storage for dynamic SQL statements used by active threads.

**Note:** For DB2 10 or later, this value is related to shared agent local variable pools above the bar.

**Field Name:** QW0225H2

**STATEMENT COUNT AT HWM**

The number of dynamic SQL local cache statements used by active threads at high storage time.

- For DB2 10 or later, this value is related to shared agent local variable pools above the bar.
- Prior to DB2 10, this value is related to the local cache storage pool below the bar.

**Field Name:** QW0225HC

**DATE AT HWM**

The timestamp at high-water storage.

**Field Name:** QW0225HT

**TIME AT HWM**

The timestamp at high-water storage.

**Field Name:** QW0225HT

**DYNAMIC STMT CACHE CNTL BLKS (MB)**

The total statement cache storage blocks above the bar (64-bit shared variable pool).

**Field Name:** QW0225S2

**SYSTEM COPIES OF CACHED SQL STMTS (MB)**

The total shareable storage allocated for dynamic SQL statements used by active threads.

- For DB2 11, this field is derived from QW0225SC and related to storage above the bar.
- Prior to DB2 11, this field is derived from QW0225SC and related to storage below the bar.
- For DB2 10, the storage is used for executable code sequences (xPROC).
- For DB2 9, it is the storage allocated for the local cache storage pool.

**Field Name:** SW0225SC

**IN USE STORAGE (MB)**
The total shareable storage requested for dynamic SQL statements used by active threads.

- For DB2 11, this field is derived from QW0225LS8 and related to storage above the bar.
- Prior to DB2 11, this field is derived from QW0225LS and related to storage below the bar.
- For DB2 10, the storage is used for executable code sequences (xPROC).
- For DB2 9, it is the storage used for thread copies in the local cache storage pool.

**Field Name:** SW0225LS

**HWM FOR ALLOCATED STATEMENTS (MB)**

A statistics interval high-water mark of requested shareable storage for dynamic SQL statements used by active threads.

- For DB2 11, this field is derived from QW0225HS8 and related to storage above the bar.
- Prior to DB2 11, this field is derived from QW0225HS and related to storage below the bar.
- For DB2 10, the storage is used for executable code sequences (xPROC).
- For DB2 9, it is the storage used for thread copies in the local cache storage pool.

**Field Name:** SW0225HS

**SYSTEM COPIES OF STATIC SQL (MB)**

The total shareable storage allocated for static SQL statements.

- For DB2 11, this field is derived from QW0225SX8 and related to storage above the bar.
- Prior to DB2 11, this field is derived from QW0225SX and used for storage of executable code sequences (xPROC) below the bar.

**Field Name:** SW0225SX

**IN USE STORAGE (MB)**

The total storage requested for shareable static SQL statements.

- Prior to DB2 11, this field is derived from QISEKSPA8 and used for storage of executable code sequences (xPROC) below the bar.
- For DB2 11, this field is derived from QISEKSPA and related to storage above the bar.

**Field Name:** SISEKSPA

**THREAD PLAN AND PACKAGE STORAGE (MB)**

The storage allocated to plans and packages above the bar.

**Field Name:** SISESQA

**ARRAY VARIABLE STORAGE**

The amount of storage in use for array variables.

**Field Name:** QW0225AR

**SHARED STORAGE MANAGER CNTL BLKS (MB)**

The amount of 64-bit shared storage allocated for storage manager control structures (DB2 field: QW0225SMS).
DBM1 Storage Above 2 GB

Field Name: SW225SMS

SHARED SYSTEM AGENT STACK STORAGE (MB)
The amount of 64-bit shared storage allocated for system agent stack use
(DB2 field: QW0225GSG_SYS).

Field Name: SW225GSY

STACK STORAGE IN USE (MB)
The amount of 64-bit shared system agent stack that is in use (DB2 field:
QW0225SUG_SYS).

Field Name: SW225SSY

SHARED NON-SYSTEM AGENT STACK STORAGE (MB)
The amount of 64-bit shared storage allocated for non-system agent stack
use (DB2 field: QW0225GSG).

Field Name: SW225GSG

STACK STORAGE IN USE (MB)
The amount of 64-bit shared non-system agent stack that is in use (DB2
field: QW0225SUG).

Field Name: SW225SUG
This topic shows detailed information about “Statistics - Dynamic SQL Statement”.

**Statistics - Dynamic SQL Statement**

The field labels shown in the following sample layout of “Statistics - Dynamic SQL Statement” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Quantity</th>
<th>/Second</th>
<th>/Thread</th>
<th>/Commit</th>
</tr>
</thead>
<tbody>
<tr>
<td>PREPARE REQUESTS</td>
<td>16.00</td>
<td>0.09</td>
<td>N/C</td>
<td>0.48</td>
</tr>
<tr>
<td>FULL PREPARES</td>
<td>2.00</td>
<td>0.01</td>
<td>N/C</td>
<td>0.06</td>
</tr>
<tr>
<td>SHORT PREPARES</td>
<td>14.00</td>
<td>0.08</td>
<td>N/C</td>
<td>0.42</td>
</tr>
<tr>
<td>GLOBAL CACHE HIT RATIO (%)</td>
<td>87.50</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>IMPLICIT PREPARES</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>PREPARES AVOIDED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>CACHE LIMIT EXCEEDED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>PREP STMT PURGED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>LOCAL CACHE HIT RATIO (%)</td>
<td>N/C</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>CSWL - STMTS PARSED</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>CSWL - LITS REPLACED</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>CSWL - MATCHES FOUND</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>CSWL - DUPLS CREATED</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>

**PREPARE REQUESTS**

The number of all Explicit and Implicit prepare requests.

An Explicit Prepare occurs when an SQL PREPARE or EXECUTE IMMEDIATE is requested by the application. An Explicit Prepare always results in either a Short Prepare or a Full Prepare.

An Implicit Prepare occurs when the user copy of the prepared SQL statement no longer exists in the local dynamic SQL cache. An Implicit Prepare always results in either a Short Prepare or a Full Prepare.

**Field Name:** SPREPSUM

**FULL PREPARES**

The number of full prepare requests.

A Full Prepare occurs for both Explicit Prepare and Implicit Prepare requests when the skeleton copy of the prepared SQL statement is not found in global dynamic SQL cache in the EDM pool.

**Field Name:** QISED5I

**SHORT PREPARES**

The number of short prepare requests.

A Short Prepare is executed for both Explicit Prepare and Implicit Prepare requests when the skeleton copy of the prepared SQL statement is found in global dynamic SQL cache in the EDM pool.

**Field Name:** SPREPSHT

**GLOBAL CACHE HIT RATIO (%)**

The ratio of successful search requests for prepared statements from the global dynamic SQL cache. This indicates the effectiveness of the global dynamic SQL cache in the EDM pool.
Dynamic SQL Statement

A value near to 100 indicates that in most cases DB2 found skeleton copies of prepared statements in global dynamic cache and could perform short prepares. A value near to 0 indicates that in most cases skeleton copies of prepared statements were not found in global dynamic cache and full prepares were performed.

Field Name: SCACHRAT
This is an exception field.

IMPLICIT PREPARES
An implicit prepare occurs when the user copy of the prepared SQL statement no longer exists in the local dynamic SQL cache and the application plan or package is bound with KEEPDYNAMIC YES.
If the skeleton copy of the prepared SQL statement exists in the global dynamic SQL cache in the EDM pool, a short prepare is executed, otherwise a full prepare is executed.

Field Name: QXSTIPRP
This is an exception field.

PREPARES AVOIDED
This field indicates the number of times where no SQL PREPARE or EXECUTE IMMEDIATE was issued by the application and a copy of a prepared SQL statement was found in local dynamic SQL cache.

When an application plan or package is bound with KEEPDYNAMIC YES, a copy of each prepared SQL statement for the application thread is held in the local dynamic SQL cache and kept across a commit boundary.

An application thread can save the total cost of a prepare by using a copy of the prepared statement in the local dynamic SQL cache from an earlier prepare by the same thread. To do this, the application must be modified to avoid issuing repetitive SQL PREPAREs for the same SQL statement.

Field Name: QXSTNPRP
This is an exception field.

CACHE LIMIT EXCEEDED
The number of times statements are invalidated in the local dynamic SQL cache because the MAXKEEPD limit has been reached and prepared SQL statements in the local dynamic SQL cache have to be reclaimed.

Field Name: QXSTDEXP
This is an exception field.

PREP STMT PURGED
The number of times statements are invalidated in the local dynamic SQL cache because of SQL DDL or updated RUNSTATS information and prepared SQL statements in the local dynamic SQL cache have to be reclaimed.

Field Name: QXSTDINV
This is an exception field.

LOCAL CACHE HIT RATIO (%)
**Dynamic SQL Statement**

The local cache hit ratio. This shows the percentage of SQL statements that avoided prepares because the statements were retrieved from the local cache. It indicates the effectiveness of the local SQL statement cache.

A value near to 100 indicates that in most cases DB2 found skeleton copies of prepared statements in local dynamic cache and avoided statement prepares.

A value near to 0 indicates that in most cases skeleton copies of prepared statements were not found in local dynamic cache and implicit prepares were performed.

**Field Name:** SLCACRAT

**CSWL - STMTS PARSED**

The number of times DB2 parsed dynamic statements because CONCENTRATE STATEMENTS WITH LITERALS behavior was used for the prepare of the statement for the dynamic statement cache.

**Field Name:** QXSTCWLP

**CSWL - LITS REPLACED**

The number of times DB2 replaced at least one literal in a dynamic statement because CONCENTRATE STATEMENTS WITH LITERALS was used for the prepare of the statement for dynamic statement cache.

**Field Name:** QXSTCWLR

**CSWL - MATCHES FOUND**

The number of times DB2 found a matching reusable copy of a dynamic statement in cache during prepare of a statement that had literals replaced because of CONCENTRATE STATEMENTS WITH LITERALS.

**Field Name:** QXSTCWLM

**CSWL - DUPLS CREATED**

The number of times DB2 created a duplicate STMT instance in the statement cache for a dynamic statement that had literals replaced by CONCENTRATE STATEMENTS WITH LITERALS behavior. The duplicate STMT instance was needed because a cache match failed because the literal reusability criteria was not met.

**Field Name:** QXSTCWLD
This topic shows detailed information about “Statistics - DB2 API”.

**Statistics - DB2 API**

The field labels shown in the following sample layout of “Statistics - DB2 API” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ABENDS</strong></td>
<td>The number of instrumentation facility interface (IFI) abends. Field Name: QWSDSCA</td>
</tr>
<tr>
<td><strong>UNRECOGNIZED</strong></td>
<td>The number of calls made to IFI using a function that is not recognized by the interface. Field Name: QWSDSCU</td>
</tr>
<tr>
<td><strong>COMMAND REQUESTS</strong></td>
<td>The number of calls made to IFI using the COMMAND function. Field Name: QWSDSCCO</td>
</tr>
<tr>
<td><strong>READA REQUESTS</strong></td>
<td>The number of calls made to IFI using the READA (read asynchronous data) function. Field Name: QWSDSCRA</td>
</tr>
<tr>
<td><strong>READS REQUESTS</strong></td>
<td>The number of calls made to IFI using the READS (read synchronous data) function. Field Name: QWSDSCRS</td>
</tr>
<tr>
<td><strong>WRITE REQUESTS</strong></td>
<td>The number of calls made to IFI using the WRITE function. Field Name: QWSDSCWR</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td>The total number of calls made to IFI. Field Name: SDIFITOT</td>
</tr>
</tbody>
</table>
This topic shows detailed information about “Statistics - DB2 Commands”.

Statistics - DB2 Commands

The field labels shown in the following sample layout of “Statistics - DB2 Commands” are described in the following section.

<table>
<thead>
<tr>
<th>DB2 COMMANDS</th>
<th>QUANTITY</th>
<th>/SECOND</th>
</tr>
</thead>
<tbody>
<tr>
<td>DISPLAY DATABASE</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DISPLAY THREAD</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DISPLAY UTILITY</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DISPLAY TRACE</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DISPLAY RLIMIT</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DISPLAY LOCATION</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DISPLAY ARCHIVE</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DISPLAY BUFFERPOOL</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DISPLAY GROUPBUFFERPOOL</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DISPLAY GROUP</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DISPLAY PROCEDURE</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DISPLAY FUNCTION</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DISPLAY LOG</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DISPLAY DDF</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DISPLAY PROFILE</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DISPLAY ACCEL</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>ALTER BUFFERPOOL</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>ALTER GROUPBUFFERPOOL</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>ALTER UTILITY</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>START DATABASE</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>START TRACE</td>
<td>1.00</td>
<td>0.01</td>
</tr>
<tr>
<td>START DB2</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>START RLIMIT</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>START DDF</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>START PROCEDURE</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>START FUNCTION</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>START PROFILE</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>START ACCEL</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>STOP DATABASE</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>STOP TRACE</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>STOP DB2</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>STOP RLIMIT</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>STOP DDF</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>STOP PROCEDURE</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>STOP FUNCTION</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>STOP PROFILE</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>STOP ACCEL</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DB2 COMMANDS CONTINUED</th>
<th>QUANTITY</th>
<th>/SECOND</th>
</tr>
</thead>
<tbody>
<tr>
<td>MODIFY TRACE</td>
<td>1.00</td>
<td>0.01</td>
</tr>
<tr>
<td>MODIFY DDF</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>CANCEL THREAD</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>TERM UTILITY</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>RECOVER BSDS</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>RECOVER INDOUBT</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>RESET INDOUBT</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>RESET GENERICLU</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>
**DB2 Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Q9STCTR0</th>
<th>Q9STCTR1</th>
<th>Q9STCTR2</th>
<th>Q9STCTRC</th>
<th>Q9STCTRG</th>
<th>Q9STCTRL</th>
<th>Q9STCTRQ</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARCHIVE LOG</td>
<td>0.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SET ARCHIVE</td>
<td>0.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SET LOG</td>
<td>0.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SET SYSPARM</td>
<td>0.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ACCESS DATABASE</td>
<td>0.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UNRECOGNIZED COMMANDS</td>
<td>0.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TOTAL</td>
<td>2.00</td>
<td>0.01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**DISPLAY DATABASE**

The number of DB2 DISPLAY DATABASE commands issued to view objects within one or more DB2 databases. This includes normal and abnormal completion of the command.

Field Name: Q9STCTR0

**DISPLAY THREAD**

The number of DB2 DISPLAY THREAD commands issued to view threads active within the DB2 subsystem. This includes normal and abnormal completion of the command.

Field Name: Q9STCTR1

**DISPLAY UTILITY**

The number of DB2 DISPLAY UTILITY commands issued to view the status of one or more DB2 utilities. This includes normal and abnormal completion of the command.

Field Name: Q9STCTR2

**DISPLAY TRACE**

The number of DB2 DISPLAY TRACE commands issued to determine the currently active DB2 traces. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRC

**DISPLAY RLIMIT**

The number of DB2 DISPLAY RLIMIT commands issued to view the current status of the DB2 resource limit facility. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRG

**DISPLAY LOCATION**

The number of DB2 DISPLAY LOCATION commands issued to display statistics about threads with a distributed relationship. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRL

**DISPLAY ARCHIVE**

The number of DB2 DISPLAY ARCHIVE commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRQ

**DISPLAY BUFFERPOOL**
DB2 Commands

The number of DB2 DISPLAY BUFFERPOOL commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRO

DISPLAY GROUPBUFFERPOOL

The number of DB2 DISPLAY GROUPBUFFERPOOL commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRT

DISPLAY GROUP

The number of DB2 DISPLAY GROUP commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRL

DISPLAY PROCEDURE

The number of DB2 DISPLAY PROCEDURE commands executed. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRZ

DISPLAY FUNCTION

The number of DB2 DISPLAY FUNCTION commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTX3

DISPLAY LOG

The number of DB2 DISPLAY LOG commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTX5

DISPLAY DDF

The number of DB2 DISPLAY DDF commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTS

DISPLAY PROFILE

The number of DB2 DISPLAY PROFILE commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTX5

DISPLAY ACCEL

The number of DB2 DISPLAY ACCEL commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTDA

ALTER BUFFERPOOL

The number of DB2 ALTER BUFFERPOOL commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTR

ALTER GROUPBUFFERPOOL
DB2 Commands

The number of DB2 ALTER GROUPBUFFERPOOL commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRS

ALTER UTILITY

The number of DB2 ALTER UTILITY commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRY

START DATABASE

The number of DB2 START DATABASE commands issued to make a database available for use. This includes normal and abnormal completion of the command.

Field Name: Q9STCTR5

START TRACE

The number of DB2 START TRACE commands issued to initiate a DB2 trace. This includes normal and abnormal completion of the command.

Field Name: Q9STCTR6

START DB2

The number of DB2 START DB2 commands issued to bring up a DB2 subsystem. This includes normal and abnormal completion of the command.

Field Name: Q9STCTR7

START RLIMIT

The number of DB2 START RLIMIT commands issued to enable the DB2 resource limit facility. This includes normal and abnormal completion of the command.

Field Name: Q9STCTR8

START DDF

The number of DB2 START DDF commands issued to enable the DB2 distributed data facility. This includes normal and abnormal completion of the command.

Field Name: Q9STCTR1

START PROCEDURE

The number of DB2 START PROCEDURE commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRV

START FUNCTION

The number of DB2 START FUNCTION commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTX0

START PROFILE

The number of DB2 START PROFILE commands issued. This includes normal and abnormal completion of the command.
DB2 Commands

Field Name: Q9STCTSS

START ACCEL
The number of DB2 START ACCEL commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTSA

STOP DATABASE
The number of DB2 STOP DATABASE commands issued to prevent access to a DB2 database. This includes normal and abnormal completion of the command.

Field Name: Q9STCTR8

STOP TRACE
The number of DB2 STOP TRACE commands issued to terminate one or more active DB2 traces. This includes normal and abnormal completion of the command.

Field Name: Q9STCTR9

STOP DB2
The number of DB2 STOP DB2 commands issued to terminate the DB2 subsystem. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRA

STOP RLIMIT
The number of DB2 STOP RLIMIT commands issued to disable the DB2 resource limit facility. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRJ

STOP DDF
The number of DB2 STOP DDF commands issued to disable the DB2 distributed data facility. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRJ

STOP PROCEDURE
The number of DB2 STOP PROCEDURE commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRW

STOP FUNCTION
The number of DB2 STOP FUNCTION commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTX1

STOP PROFILE
The number of DB2 STOP PROFILE commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTST
DB2 Commands

STOP ACCEL
The number of DB2 STOP ACCEL commands issued. This includes normal and abnormal completion of the command.
Field Name: Q9STCTXA

MODIFY TRACE
The number of DB2 MODIFY TRACE commands issued to alter trace events (IFCIDs) for an active trace. This includes normal and abnormal completion of the command.
Field Name: Q9STCTRTH

MODIFY DDF
The number of DB2 MODIFY DDF commands issued. This includes normal and abnormal completion of the command.
Field Name: Q9STCDMD

CANCEL THREAD
The number of DB2 CANCEL THREAD commands issued to cancel a thread. This includes normal and abnormal completion of the command.
Field Name: Q9STCTRK

TERM UTILITY
The number of DB2 TERM UTILITY commands issued to stop execution of a DB2 utility. This includes normal and abnormal completion of the command.
Field Name: Q9STCTRB

RECOVER BSDS
The number of DB2 RECOVER BSDS commands issued to reestablish dual bootstrap data sets after one has been disabled by a data set error. This includes normal and abnormal completion of the command.
Field Name: Q9STCTR3

RECOVER INDOUBT
The number of DB2 RECOVER INDOUBT commands issued to recover threads left indoubt because DB2 or a transaction manager could not automatically recover them. This includes normal and abnormal completion of the command.
Field Name: Q9STCTR4

RESET INDOUBT
The number of DB2 RESET INDOUBT commands issued. This includes normal and abnormal completion of the command.
Field Name: Q9STCTRR

RESET GENERICLU
The number of DB2 RESET GENERICLU commands issued. This includes normal and abnormal completion of the command.
Field Name: Q9STCTR5

ARCHIVE LOG
DB2 Commands

The number of DB2 ARCHIVE LOG commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTR M

SET ARCHIVE

The number of DB2 SET ARCHIVE commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTRP

SET LOG

The number of DB2 SET LOG commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTX2

SET SYSPARM

The number of DB2 SET SYSPARM commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTX4

ACCESS DATABASE

The number of DB2 ACCESS DATABASE commands issued. This includes normal and abnormal completion of the command.

Field Name: Q9STCTAD

UNRECOGNIZED COMMANDS

The number of commands not recognized by DB2. The number is incremented if the command verb or primary keyword cannot be determined. For example:

- "-DISPLOX DATABASE(*)" is an unknown verb.
- "-DISPLAY FATAFASE(*)" is an unknown primary keyword.

Field Name: Q9STEROR

TOTAL

The total number of DB2 commands that were issued.

Field Name: SDSTTOTL
This topic shows detailed information about “Statistics - DIST Storage Above 2 GB (DB2 10 or later)”.

**Statistics - DIST Storage Above 2 GB (DB2 10 or later)**

The field labels shown in the following sample layout of “Statistics - DIST Storage Above 2 GB (DB2 10 or later)” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>QW0225GA</td>
<td>FIXED STORAGE (MB)</td>
</tr>
<tr>
<td>QW0225FA</td>
<td>GETMAINED STORAGE (MB)</td>
</tr>
<tr>
<td>QW0225VA</td>
<td>VARIABLE STORAGE (MB)</td>
</tr>
<tr>
<td>QW0225SM</td>
<td>STORAGE MANAGER CONTROL BLOCKS (MB)</td>
</tr>
</tbody>
</table>

**GETMAINED STORAGE (MB)**

Total storage acquired by GETMAIN. This includes space for the compression dictionary, and statement and DBD cache that can be used by the Environmental Descriptor Manager (EDM).

This figure can be different from the sum of GETMAIN storage items shown in the statistics DBM1 storage, because DB2 does not produce grouping statistics for all GETMAIN storage.

Field Name: QW0225GA

**FIXED STORAGE (MB)**

The total amount of fixed storage above the 2 GB bar.

Field Name: QW0225FA

**VARIABLE STORAGE (MB)**

Amount of variable storage available above the 2 GB bar.

Field Name: QW0225VA

**STORAGE MANAGER CONTROL BLOCKS (MB)**

Total 64-bit storage allocated for storage manager control structures.

Field Name: QW0225SM
DIST and MVS Storage Below 2 GB (DB2 10 or later)

This topic shows detailed information about “Statistics - DIST and MVS Storage Below 2 GB (DB2 10 or later)”. 

Statistics - DIST and MVS Storage Below 2 GB (DB2 10 or later)

The field labels shown in the following sample layout of “Statistics - DIST and MVS Storage Below 2 GB (DB2 10 or later)” are described in the following section.

DIST AND MVS STORAGE BELOW 2 GB QUANTITY
-------------------------------------------- ------------------
TOTAL DIST STORAGE BELOW 2 GB (MB) 133.03
TOTAL GETMAIN STORAGE (MB) 0.04
TOTAL VARIABLE STORAGE (MB) 14.21
NUMBER OF ACTIVE CONNECTIONS 967.74
NUMBER OF INACTIVE CONNECTIONS 0.00
TOTAL FIXED STORAGE (MB) 0.85
TOTAL GETMAIN STACK STORAGE (MB) 117.92
TOTAL STACK STORAGE IN USE (MB) 117.89
SYSTEM AGENT STACK STORAGE IN USE (MB) 15.73
STORAGE CUSHION (MB) 358.03
24 BIT LOW PRIVATE (MB) 0.23
24 BIT HIGH PRIVATE (MB) 0.21
24 BIT PRIVATE CURRENT HIGH ADDRESS 0000000000042000
31 BIT EXTENDED LOW PRIVATE (MB) 5.14
31 BIT EXTENDED HIGH PRIVATE (MB) 147.14
31 BIT PRIVATE CURRENT HIGH ADDRESS 0000000018325000
EXTENDED REGION SIZE (MAX) (MB) 1666.00

**TOTAL DIST STORAGE BELOW 2 GB (MB)**

Total DIST storage below the bar. This includes:
- Fixed length storage use
- Getmained storage
- Save areas
- Variables

**Field Name:** SW0225DI

**TOTAL GETMAINED STORAGE (MB)**

Total storage acquired by GETMAIN. This includes space for virtual pools, EDM pool, compression dictionary, castout buffers, and the data space lookaside buffer, hiperpool control blocks, and data space buffer pool control blocks.

This figure can be different from the sum of GETMAIN storage items shown in the statistics DBM1 storage, because DB2 does not produce grouping statistics for all GETMAIN storage.

**Field Name:** QW0225GM

**TOTAL VARIABLE STORAGE (MB)**

Total storage used by all variable pools. This includes storage used by:
- System agents
- Local agents
- RID pool
- Pipe manager subpool
- Local dynamic statement cache control blocks
DIST and MVS Storage Below 2 GB

- Local dynamic statement cache statement pool
- Buffer and data manager trace tables
- A list of objects in restricted state including the new PRO state. If consumption of this storage pool is high, review restrictive exception state of database objects and check whether they can be resolved or reduced.

Field Name: QW0225VR

NUMBER OF ACTIVE CONNECTIONS
The number of active connections, or active and disconnected DBAT threads.
Field Name: SACDBATS

NUMBER OF INACTIVE CONNECTIONS
The current number of type 2 inactive connections.
Field Name: QDSTCIN2

TOTAL FIXED STORAGE (MB)
Total amount of fixed storage.
Field Name: QW0225FX

TOTAL GETMAINED STACK STORAGE (MB)
Total GETMAINED storage allocated for program stack use.
Field Name: QW0225GS

TOTAL STACK STORAGE IN USE (MB)
The amount of stack storage that is in use.
Field Name: QW0225SU

SYSTEM AGENT STACK STORAGE IN USE (MB)
The amount of 31-bit stack storage that is in use for system agents. This is a subset of QW0225SU.
Field Name: QW0225SS

STORAGE CUSHION (MB)
Storage reserved to allow DB2 to complete critical functions while short on storage. This includes the contract warning cushion, storage reserved for must-complete operations, and storage for MVS use.
Field Name: STORCUSH

24 BIT LOW PRIVATE (MB)
The amount of private MVS storage below the 16 MB line. This storage is obtained from bottom upward, usually for unauthorized programs.
Field Name: QW0225LO

24 BIT HIGH PRIVATE (MB)
The amount of private MVS storage below the 16 MB line. This storage is obtained from top downward, usually for authorized programs.
Field Name: QW0225HI

24 BIT PRIVATE CURRENT HIGH ADDRESS
DIST and MVS Storage Below 2 GB

The current high address of the 24-bit private region.

Field Name: QW0225TP

31 BIT EXTENDED LOW PRIVATE (MB)
The amount of private MVS storage above the 16 MB line. This storage is obtained from bottom upward, usually for unauthorized programs.

Field Name: QW0225EL

31 BIT EXTENDED HIGH PRIVATE (MB)
The amount of private MVS storage above the 16 MB line. This storage is obtained from top downward, usually for authorized programs.

Field Name: QW0225EH

31 BIT PRIVATE CURRENT HIGH ADDRESS
The current high address of the 31-bit private region.

Field Name: QW0225EP

EXTENDED REGION SIZE (MAX) (MB)
The maximum amount of MVS private storage available above the 16 MB line.

Field Name: QW0225RG
**DRDA Remote Locations**

This topic shows detailed information about “Statistics - DRDA Remote Locations”.

### Statistics - DRDA Remote Locations

The field labels shown in the following sample layout of “Statistics - DRDA Remote Locations” are described in the following section.

<table>
<thead>
<tr>
<th>Field Labels</th>
<th>SENT</th>
<th>RECEIVED</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRDA REMOTE LOCS SENT RECEIVED</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TRANSATIONS</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>CONVERSATIONS</td>
<td>0.00</td>
<td>2.00</td>
</tr>
<tr>
<td>CONVERSATIONS QUEUED</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>CONVERSATIONS DEALLOCATED</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>SQL STATEMENTS</td>
<td>0.00</td>
<td>49.00</td>
</tr>
<tr>
<td>(SINGLE PHASE) COMMITS</td>
<td>0.00</td>
<td>33.00</td>
</tr>
<tr>
<td>(SINGLE PHASE) ROLLBACKS</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>ROWS</td>
<td>16.00</td>
<td>0.00</td>
</tr>
<tr>
<td>MESSAGES</td>
<td>1540.00</td>
<td>1540.00</td>
</tr>
<tr>
<td>BYTES</td>
<td>140.7K</td>
<td>280.5K</td>
</tr>
<tr>
<td>BLOCKS</td>
<td>32.00</td>
<td>0.00</td>
</tr>
<tr>
<td>MESSAGES IN BUFFER</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>CONT-&gt;LIM.BLOCK FETCH SWTCH</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>STATEMENTS BOUND AT SERVER</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>PREPARE REQUEST</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>LAST AGENT REQUEST</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>TWO PHASE COMMIT REQUEST</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>TWO PHASE BACKOUT REQUEST</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>FORGET RESPONSES</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>COMMIT RESPONSES</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>BACKOUT RESPONSES</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>THREAD INDOUBT-REM.L.COORD.</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>COMMITS DONE-REM.LOC.COORD.</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>BACKOUTS DONE-REM.L.COORD.</td>
<td>N/A</td>
<td></td>
</tr>
</tbody>
</table>

**SENT - TRANSACTIONS**

The number of DBAT allocation requests sent to the remote location. This value is only meaningful at the requester location.

**Field Name:** QLSITTRNS

**SENT - CONVERSATIONS**

The number of conversations that were initiated from the requester location. This value is maintained at the requester location.

A conversation is a specific instance of using TCP/IP or SNA LU 6.2 to transfer information between a requester and a server. A conversation is a logical connection between a requester and a server.

**Field Name:** QLSTCNVS

**SENT - CONVERSATIONS QUEUED**

The number of conversation requests queued by the distributed data facility and waiting for allocation. This value is maintained at the requester location.

**Background and Tuning Information**

When this value is high, increase the limit for the number of conversations.

**Field Name:** QLSTCNVQ
This is an exception field.

**SENT - CONVERSATIONS DEALLOCATED**

The number of conversations that were deallocated from this site to the remote site.

Field Name: QLSTCNVT

**SENT - SQL STATEMENTS**

The number of SQL statements sent to the remote server. This value is updated at the requester location.

Field Name: QLSTSQLS

**SENT - (SINGLE PHASE) COMMITS**

This field depends on the DB2 version that is installed:

- **DB2 10 or later**: The number of commit requests sent to the server (single-phase commit protocol) and the committed requests sent to the participant (two-phase commit protocol).
- **Prior to DB2 10**: The number of commit requests sent to the server location (single-phase commit operations only). This value is maintained at the requester location.

Field Name: QLSTCOMS

**SENT - (SINGLE PHASE) ROLLBACKS**

This field depends on the DB2 version that is installed:

- **DB2 10 or later**: The number of abort requests sent to the server (single-phase commit protocol) and backout requests sent to the participant (two-phase commit protocol).
- **Prior to DB2 10**: The number of rollback requests sent to the remote server location (single-phase commit operations only). This value is maintained at the requester location.

Field Name: QLSTABRS

**SENT - ROWS**

The number of data rows sent to the requester location (includes SQLDA). This value is updated at the server location.

Field Name: QLSTROWS

**SENT - MESSAGES**

The number of messages sent to the remote location. A message is a group of characters and control bit sequences transferred on a single TCP/IP or SNA API call. This value is maintained at the location where the messages originated.

Field Name: QLSTMGS

**SENT - BYTES**

The number of bytes of data sent to the requester location. This value is maintained at the server location.

Field Name: QLSTBYTS

**SENT - BLOCKS**
DRDA Remote Locations

The number of blocks transmitted using block fetch. This value is maintained at the server location.

Field Name: QLSTBTBF

SENT - MESSAGES IN BUFFER

The number of rows transmitted or received in DB2 message buffers using block fetch. This field includes both requester and server activity.

Field Name: QLSTBROW

This is an exception field.

SENT - CONT->LIM.BLOCK FETCH SWTCH

The number of times the continuous block fetch was switched to a limited block fetch (DB2 private protocol only). This value is maintained at the requester location.

Background and Tuning Information

When this value is high, consider tuning VTAM.

Field Name: QLSTCBLB

This is an exception field.

SENT - STATEMENTS BOUND AT SERVER

The number of SQL statements that were bound for remote access (DB2 private protocol only). This value is maintained at the requester location.

Field Name: QLSTRBND

This is an exception field.

SENT - PREPARE REQUEST

The number of prepare requests sent to the participant (two-phase commit operations only).

Field Name: QLSTPRSE

This is an exception field.

SENT - LAST AGENT REQUEST

The number of last agent requests sent to the coordinator (two-phase commit operations only).

A last agent request reduces the number of messages to be sent for the commit. When DB2 is the requester, this number is incremented when a conversation is deallocated and the conversation was not used since the last commit.

Background and Tuning Information

If this number is large and your application design allows for it, you can store another message by issuing a release before the commit (only for a DB2 requester).

Field Name: QLSTLASE

SENT - TWO PHASE COMMIT REQUEST

The number of commit requests sent to the participant (two-phase commit operations only).

Field Name: QLSTCRSE
SENT - TWO PHASE BACKOUT REQUEST
The number of backout requests sent to the participant (two-phase commit operations only).

Field Name: QLSTBKSE

SENT - FORGET RESPONSES
The number of forget responses sent to the coordinator (two-phase commit operations only). This indicates that the participant was read-only.

Field Name: QLSTRRRSE

SENT - COMMIT RESPONSES
The number of request commit responses sent to the coordinator (two-phase commit operations only).

Field Name: QLSTVYSE

SENT - BACKOUT RESPONSES
The number of backout responses sent to the coordinator (two-phase commit operations only). This indicates that the participant voted no to the prepare request.

Field Name: QLSTVNSE

SENT - THREAD INDOUBT-REM.L.COORD.
The number of threads that became indoubt with the remote location as the coordinator (two-phase commit operations only). A large value might indicate network problems.

Field Name: QLSTINDT

SENT - COMMITS DONE-REM.LOC.COORD.
The number of commit operations performed with the remote location as the coordinator (two-phase commit operations only).

Field Name: QLSTCPTR

SENT - BACKOUTS DONE-REM.L.COORD.
The number of rollback operations performed with the remote location as the coordinator (two-phase commit operations only).

Field Name: QLSTRBTR

RECEIVED - TRANSACTIONS
The number of DBAT allocation requests received from the remote location. This value is only meaningful at the server location.

Field Name: QLSTRBR

RECEIVED - CONVERSATIONS
The number of conversations that were initiated from the requester to the server location. This value is updated at the server location.

Field Name: QLSTCNVR

RECEIVED - SQL STATEMENTS
The number of SQL statements received from the requester location. This value is updated at the server location.
Field Name: QLSTSQRL

RECEIVED - (SINGLE PHASE) COMMITS

This field depends on the DB2 version that is installed:

- **DB2 10 or later**: The number of commit requests received from the requester (single-phase commit protocol) and committed requests received from the coordinator (two-phase commit protocol).
- **Prior to DB2 10**: The number of commit requests received from the requester location (single-phase commit operations only). This value is maintained at the server location.

Field Name: QLSTCOMR

RECEIVED - (SINGLE PHASE) ROLLBACKS

This field depends on the DB2 version that is installed:

- **DB2 10 or later**: The number of abort requests received from the requester (single-phase commit protocol) and backout requests received from the coordinator (two-phase commit protocol).
- **Prior to DB2 10**: The number of rollback requests received from the requester location (single-phase commit operations only). This value is maintained at the server location.

Field Name: QLSTABRR

RECEIVED - ROWS

The number of data rows received from the server location. This value is maintained at the requester location.

Note:

- This value does not include any SQLDA or SQLCA transmitted.
- Block fetch can significantly affect the number of rows sent across the network. When used with nonupdate cursors, block fetch groups as many rows as possible into the message buffer, and transmits the buffer over the network without requiring a VTAM message. Consequently, more rows of data might be sent from the server location than are received by the requester location.

This is especially true when DB2 private protocol is used because multiple blocks can be transmitted from the server with no intervening messages from the requester.

Field Name: QLSTROWR

RECEIVED - MESSAGES

The number of messages received by VTAM from the remote location. This value is maintained at the location where the messages were received.

More messages might be sent from the server location than are received by the requester due to the manner in which distributed SQL statements are processed internally.

Field Name: QLSTMGR

RECEIVED - BYTES

The number of bytes of data received from the server location. This value is maintained at the requester location.
More bytes of data might be sent from the server location than are received by the requester due to the manner in which distributed SQL statements are processed internally.

Field Name: QLSTBYTR

RECEIVED - BLOCKS

The number of blocks received from the remote location using block fetch. This value is maintained at the requester location.

Field Name: QLSTBRBF

RECEIVED - PREPARE REQUEST

The number of prepare requests received from the coordinator (two-phase commit operations only).

Field Name: QLSTPRRC

RECEIVED - LAST AGENT REQUEST

The number of last agent requests received from the initiator (two-phase commit operations only).

This number is incremented when the DB2 server is receiving a last agent request from its requester.

Field Name: QLSTLARC

RECEIVED - TWO PHASE COMMIT REQUEST

The number of commit requests received from the coordinator (two-phase commit operations only).

Field Name: QLSTCRRC

RECEIVED - TWO PHASE BACKOUT REQUEST

The number of backout requests received from the coordinator (two-phase commit operations only).

Field Name: QLSTBKRC

RECEIVED - FORGET RESPONSES

The number of forget responses received from the participant (two-phase commit operations only). This indicates that the participant was read-only.

Field Name: QLSTFRRC

RECEIVED - COMMIT RESPONSES

The number of request commit responses received from the participant (two-phase commit operations only).

Field Name: QLSTCRRC

RECEIVED - BACKOUT RESPONSES

The number of backout responses received from the participant (two-phase commit operations only). This indicates that the participant voted no to the prepare request.

Field Name: QLSTVRRC
EDM Pool Activity

This topic shows the report and trace blocks for EDM Pool Activity that are provided for the different versions of DB2.

- “EDM Pool Activity (DB2 9)” on page 49-87
  This topic shows detailed information about “Statistics - EDM Pool Activity (DB2 9)”.

- “EDM Pool Activity (DB2 10 or later)” on page 49-93
  This topic shows detailed information about “Statistics - EDM Pool Activity (DB2 10 or later)”.

OMEGAMON XE for DB2 PE & PM: Report Reference
EDM Pool Activity (DB2 9)

This topic shows detailed information about “Statistics - EDM Pool Activity (DB2 9)”.

This report block shows data for the EDM:
- RDS pool below the 2 GB bar: part of a plan (CT) or package (PT) that is below the 2 GB bar
- RDS pool above the 2 GB bar: part of a plan (CT) or package (PT) that is above the 2 GB bar
- DBD cache above the 2 GB bar
- Statement Cache above the 2 GB bar
- Skeleton pool above the 2 GB bar: skeleton copies of plans (SKCTs) and packages (SKPTs).

Statistics - EDM Pool Activity (DB2 9)

The field labels shown in the following sample layout of “Statistics - EDM Pool Activity (DB2 9)” are described in the following section.

<table>
<thead>
<tr>
<th>EDM POOL</th>
<th>QUANTITY</th>
<th>/SECOND</th>
<th>/THREAD</th>
<th>/COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>PAGES IN RDS POOL (BELOW)</td>
<td>25000.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>HELD BY CT</td>
<td>6.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>HELD BY PT</td>
<td>0.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>FREE PAGES</td>
<td>24994.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>FAILS DUE TO POOL FULL</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>PAGES IN RDS POOL (ABOVE)</td>
<td>524.3K</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>HELD BY CT</td>
<td>0.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>HELD BY PT</td>
<td>0.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>FREE PAGES</td>
<td>524.3K</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>FAILS DUE TO RDS POOL FULL</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>PAGES IN DBD POOL (ABOVE)</td>
<td>25600.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>HELD BY DBD</td>
<td>142.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>FREE PAGES</td>
<td>25458.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>FAILS DUE TO DBD POOL FULL</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>PAGES IN STMT POOL (ABOVE)</td>
<td>25600.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>HELD BY STATEMENTS</td>
<td>7802.43</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>FREE PAGES</td>
<td>17797.57</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>FAILS DUE TO STMT POOL FULL</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>PAGES IN SKEL POOL (ABOVE)</td>
<td>25600.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>HELD BY SKCT</td>
<td>7.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>HELD BY SKPT</td>
<td>640.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>FREE PAGES</td>
<td>24953.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>FAILS DUE TO SKEL POOL FULL</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DBD REQUESTS</td>
<td>22444.00</td>
<td>6.74</td>
<td>1496.27</td>
<td>1.83</td>
</tr>
<tr>
<td>DBD NOT FOUND</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DBD HIT RATIO (%)</td>
<td>100.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>CT REQUESTS</td>
<td>3.00</td>
<td>0.00</td>
<td>0.20</td>
<td>0.00</td>
</tr>
<tr>
<td>CT NOT FOUND</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>CT HIT RATIO (%)</td>
<td>100.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>PT REQUESTS</td>
<td>101.5K</td>
<td>30.49</td>
<td>6765.93</td>
<td>8.26</td>
</tr>
<tr>
<td>PT NOT FOUND</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>PT HIT RATIO (%)</td>
<td>100.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>PKG SEARCH NOT FOUND</td>
<td>3.00</td>
<td>0.00</td>
<td>0.20</td>
<td>0.00</td>
</tr>
<tr>
<td>PKG SEARCH NOT FOUND INSERT</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>
**EDM Pool Activity (DB2 9)**

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PKG SEARCH NOT FOUND DELETE</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>STATEMENTS IN GLOBAL CACHE</td>
<td>2507.48 N/A N/A N/A</td>
</tr>
<tr>
<td>PAGES IN RDS POOL (BELOW)</td>
<td></td>
</tr>
<tr>
<td>HELD BY CT</td>
<td>The current number of pages used for the cursor tables (CTs). This is a snapshot value.</td>
</tr>
<tr>
<td>HELD BY PT</td>
<td>The current number of pages used for package tables (PTs). This is a snapshot value.</td>
</tr>
<tr>
<td>FREE PAGES</td>
<td>The number of pages currently not used by any object in the EDM pool, in the EDM pool (below), or in the RDS pool (below). This is a snapshot value.</td>
</tr>
<tr>
<td>FAILS DUE TO POOL FULL</td>
<td>The total number of failures because the EDM pool or EDM pool (below) was full.</td>
</tr>
<tr>
<td>PAGES IN RDS POOL (ABOVE)</td>
<td>The number of pages in the RDS pool above the 2 GB bar.</td>
</tr>
</tbody>
</table>

**Field Name** QISEPAGE

**Field Name** QISECT

**Field Name** QISEKT

**Field Name** QISEFREE

**Field Name** QISEFAIL

This is an exception field.

This is an exception field.

**Field Name** QISESPGE

**Field Name** QISECTA

**Field Name** QISEKTA

**Field Name** QISESFRE
FAILS DUE TO RDS POOL FULL
The number of failures because the RDS pool above the 2 GB bar was full.

Field Name: QISESFAL

PAGES IN DBD POOL (ABOVE)
This field shows the number of pages in the DBD pool above the 2 GB bar.

Field Name: QISEDPGE

HELD BY DBD
The current number of pages used for database descriptors (DBDs). This is a snapshot value.

Field Name: QISEDBD

FREE PAGES
This field shows the number of free pages in the DBD pool above the 2 GB bar.

Field Name: QISEDFRE
This is an exception field.

FAILS DUE TO DBD POOL FULL
This field shows the total number of failures because the DBD pool above the 2 GB bar was full.

Field Name: QISEDFAL
This is an exception field.

PAGES IN STMT POOL (ABOVE)
The current number of pages in the EDM Statement pool above the 2 GB bar. This is a snapshot value.

Field Name: QISECPGE

HELD BY STATEMENTS
The number of pages in the EDM Statement pool above the 2 GB bar that is used for cached dynamic SQL statements. This is a snapshot value.

Field Name: QISEDYNP

FREE PAGES
The number of pages currently not used by any object in the EDM Statement pool above the 2 GB bar.

Field Name: QISECFRE

FAILS DUE TO STMT POOL FULL
The total number of failures because the EDM Statement pool above the 2 GB bar was full.

Field Name: QISECFAL

PAGES IN SKEL POOL (ABOVE)
The current number of pages in the EDM skeleton pool above the 2 GB bar.

Field Name: QISEKPGE
EDM Pool Activity (DB2 9)

HELD BY SKCT
The current number of pages used for skeleton cursor tables (SKCTs). This is a snapshot value.
Field Name: QISESKCT

HELD BY SKPT
The current number of pages used for skeleton package tables (SKPTs). This is a snapshot value.
Field Name: QISESKPT

FREE PAGES
The number of pages currently not used by any object in the EDM skeleton pool above the 2 GB bar.
Field Name: QISEKFRE

FAILS DUE TO SKEL POOL FULL
The total number of failures because the EDM skeleton pool above the 2 GB bar was full.
Field Name: QISEKFAL

DBD REQUESTS
The number of requests for database descriptors (DBDs).
Field Name: QISEDBDG

DBD NOT FOUND
The total number of times database descriptors were loaded from DASD.
To find the number of times the DBD was already in the EDM pool, subtract this value from the value of Requests for sections - DBD field.
Field Name: QISEDBDL
This is an exception field.

DBD HIT RATIO (%)
The ratio of successful requests for database descriptors (DBD) from the EDM pool to the total number of requests for database descriptors expressed as a percentage.
Field Name: SERDBLR
This is an exception field.

CT REQUESTS
The number of requests for cursor table (CT) sections.
Field Name: QISECTG
This is an exception field.

CT NOT FOUND
The number of times a cursor table section was loaded from DASD.
To find the number of times the CT was found in the EDM pool, subtract this value from the value of the Requests for sections - CT field.
Field Name: QISECTL
This is an exception field.

**CT HIT RATIO (%)**

The ratio of successful requests for cursor tables from the EDM pool to the total number of requests for cursor tables expressed as a percentage.

**Field Name:** SERCTLR
This is an exception field.

**PT REQUESTS**

The number of requests for package table (PT) sections.

**Field Name:** QISEKTG
This is an exception field.

**PT NOT FOUND**

The number of times a package table section was loaded from DASD.
To find the number of times the PT was already in the EDM pool, subtract this value from the value of the Requests for sections - PT field.

**Field Name:** QISEKTL
This is an exception field.

**PT HIT RATIO (%)**

The ratio of successful package table requests from the EDM pool to the total number of package table requests, expressed as a percentage.

**Field Name:** SERPTLR
This is an exception field.

**PKG SEARCH NOT FOUND**

When a package is bound with a wild card (*) for package names, in the form of PKLIST(COL1.*,COL2.*,.....), EDM generates a NOT-FOUND record to avoid future I/O if a collection ID/package name combination does not exist.

This field shows how often a cached record was located during package binding.

**Field Name:** QISEKNFM

**PKG SEARCH NOT FOUND INSERT**

When a package is bound with a wild card (*) for package names, in the form of PKLIST(COL1.*,COL2.*,.....), EDM generates a NOT-FOUND record to avoid future I/O if a collection ID/package name combination does not exist.

This field shows how often a record was added to the cache during package binding.

**Field Name:** QISEKNFA

**PKG SEARCH NOT FOUND DELETE**

When a package is bound with a wild card (*) for package names, in the form of PKLIST(COL1.*,COL2.*,.....), EDM generates a NOT-FOUND record to avoid future I/O if a collection ID/package name combination does not exist.
EDM Pool Activity (DB2 9)

This field shows how often a record was removed from the cache during package binding.

Field Name: QISEKNFR

STATEMENTS IN GLOBAL CACHE

Number of statements in the global cache.

Field Name: QISESTMT
EDM Pool Activity (DB2 10 or later)

This topic shows detailed information about “Statistics - EDM Pool Activity (DB2 10 or later)”.

This report block shows data for the EDM pool activity for DB2 10.

Note: New ratios % PAGES IN USE (based on the stealable values) are shown, calculated as 100 - (((stealable+free)/total)*100).

Statistics - EDM Pool Activity (DB2 10 or later)

The field labels shown in the following sample layout of “Statistics - EDM Pool Activity (DB2 10 or later)” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>QISEDPGE</td>
<td>Pages in DBD pool (above)</td>
</tr>
<tr>
<td>QISEDBD</td>
<td>Held by DBD</td>
</tr>
<tr>
<td></td>
<td>Held by statements</td>
</tr>
<tr>
<td></td>
<td>Held by skct</td>
</tr>
<tr>
<td></td>
<td>Held by skpt</td>
</tr>
<tr>
<td></td>
<td>Stealable pages</td>
</tr>
<tr>
<td></td>
<td>Free pages</td>
</tr>
<tr>
<td></td>
<td>% pages in use</td>
</tr>
<tr>
<td></td>
<td>Fails due to DBD pool full</td>
</tr>
<tr>
<td></td>
<td>Pages in stmt pool (above)</td>
</tr>
<tr>
<td></td>
<td>Held by statements</td>
</tr>
<tr>
<td></td>
<td>Free pages</td>
</tr>
<tr>
<td></td>
<td>Fails due to stmt pool full</td>
</tr>
<tr>
<td></td>
<td>Pages in skel pool (above)</td>
</tr>
<tr>
<td></td>
<td>Held by skct</td>
</tr>
<tr>
<td></td>
<td>Held by skpt</td>
</tr>
<tr>
<td></td>
<td>Stealable pages</td>
</tr>
<tr>
<td></td>
<td>Free pages</td>
</tr>
<tr>
<td></td>
<td>% pages in use</td>
</tr>
<tr>
<td></td>
<td>Fails due to skel pool full</td>
</tr>
<tr>
<td></td>
<td>DBD requests</td>
</tr>
<tr>
<td></td>
<td>DBD not found</td>
</tr>
<tr>
<td></td>
<td>DBD hit ratio (%)</td>
</tr>
<tr>
<td></td>
<td>CT requests</td>
</tr>
<tr>
<td></td>
<td>CT not found</td>
</tr>
<tr>
<td></td>
<td>CT hit ratio (%)</td>
</tr>
<tr>
<td></td>
<td>PT requests</td>
</tr>
<tr>
<td></td>
<td>PT not found</td>
</tr>
<tr>
<td></td>
<td>PT hit ratio (%)</td>
</tr>
<tr>
<td></td>
<td>PKG search not found</td>
</tr>
<tr>
<td></td>
<td>PKG search not found insert</td>
</tr>
<tr>
<td></td>
<td>PKG search not found delete</td>
</tr>
<tr>
<td></td>
<td>Statements in global cache</td>
</tr>
</tbody>
</table>

PAGES IN DBD POOL (ABOVE)

This field shows the number of pages in the DBD pool above the 2 GB bar.

Field Name: QISEDPGE

HELD BY DBD

The current number of pages used for database descriptors (DBDs). This is a snapshot value.

Field Name: QISEDBD

STEALABLE PAGES
EDM Pool Activity

The current number of stealable pages used for database descriptors (DBDs).

Field Name: QISEDLRU

FREE PAGES

This field shows the number of free pages in the DBD pool above the 2 GB bar.

Field Name: QISEDFRE

This is an exception field.

% PAGES IN USE

The percentage of DBD pages in use expressed as complement of the percentage of available DBD pages (ratio of stealable and free pages to the total number).

Field Name: SISEDPIU

FAILS DUE TO DBD POOL FULL

This field shows the total number of failures because the DBD pool above the 2 GB bar was full.

Field Name: QISEDFAL

This is an exception field.

PAGES IN STMT POOL (ABOVE)

The current number of pages in the EDM Statement pool above the 2 GB bar. This is a snapshot value.

Field Name: QISECPGE

HELD BY STATEMENTS

The number of pages in the EDM Statement pool above the 2 GB bar that is used for cached dynamic SQL statements. This is a snapshot value.

Field Name: QISEDYNP

FREE PAGES

The number of pages currently not used by any object in the EDM Statement pool above the 2 GB bar.

Field Name: QISECFRE

FAILS DUE TO STMT POOL FULL

The total number of failures because the EDM Statement pool above the 2 GB bar was full.

Field Name: QISECFAL

PAGES IN SKEL POOL (ABOVE)

The current number of pages in the EDM skeleton pool above the 2 GB bar.

Field Name: QISEKPGE

HELD BY SKCT

The current number of pages used for skeleton cursor tables (SKCTs). This is a snapshot value.
**EDM Pool Activity**

**Field Name:** QISESKCT

**HELD BY SKPT**

The current number of pages used for skeleton package tables (SKPTs).
This is a snapshot value.

**Field Name:** QISESKPT

**STEALABLE PAGES**

The current number of stealable pages used for skeleton cursor and package tables.

**Field Name:** QISEKLRU

**FREE PAGES**

The number of pages currently not used by any object in the EDM skeleton pool above the 2 GB bar.

**Field Name:** QISEKFRE

**% PAGES IN USE**

The percentage of skeleton pages in use expressed as complement of the percentage of available skeleton pages (ratio of stealable and free pages to the total number).

**Field Name:** SISEKPIU

**FAILS DUE TO SKEL POOL FULL**

The total number of failures because the EDM skeleton pool above the 2 GB bar was full.

**Field Name:** QISEKFAL

**DBD REQUESTS**

The number of requests for database descriptors (DBDs).

**Field Name:** QISEDBDG

**DBD NOT FOUND**

The total number of times database descriptors were loaded from DASD.
To find the number of times the DBD was already in the EDM pool, subtract this value from the value of Requests for sections - DBD field.

**Field Name:** QISEDBDL

This is an *exception* field.

**DBD HIT RATIO (%)**

The ratio of successful requests for database descriptors (DBD) from the EDM pool to the total number of requests for database descriptors expressed as a percentage.

**Field Name:** SERDBLR

This is an *exception* field.

**CT REQUESTS**

The number of requests for cursor table (CT) sections.

**Field Name:** QISECTG
EDM Pool Activity

This is an *exception* field.

**CT NOT FOUND**

The number of times a cursor table section was loaded from DASD.

To find the number of times the CT was found in the EDM pool, subtract this value from the value of the Requests for sections - CT field.

*Field Name:* QISECTL

This is an *exception* field.

**CT HIT RATIO (%)**

The ratio of successful requests for cursor tables from the EDM pool to the total number of requests for cursor tables expressed as a percentage.

*Field Name:* SERCTLR

This is an *exception* field.

**PT REQUESTS**

The number of requests for package table (PT) sections.

*Field Name:* QISEKTG

This is an *exception* field.

**PT NOT FOUND**

The number of times a package table section was loaded from DASD.

To find the number of times the PT was already in the EDM pool, subtract this value from the value of the Requests for sections - PT field.

*Field Name:* QISEKTL

This is an *exception* field.

**PT HIT RATIO (%)**

The ratio of successful package table requests from the EDM pool to the total number of package table requests, expressed as a percentage.

*Field Name:* SERPTLR

This is an *exception* field.

**PKG SEARCH NOT FOUND**

When a package is bound with a wild card (*) for package names, in the form of PKLIST(COL1.*,COL2.*.....), EDM generates a NOT-FOUND record to avoid future I/O if a collection ID/package name combination does not exist.

This field shows how often a cached record was located during package binding.

*Field Name:* QISEKNFM

**PKG SEARCH NOT FOUND INSERT**

When a package is bound with a wild card (*) for package names, in the form of PKLIST(COL1.*,COL2.*.....), EDM generates a NOT-FOUND record to avoid future I/O if a collection ID/package name combination does not exist.
This field shows how often a record was added to the cache during package binding.

**Field Name:** QISEKNFA

**PKG SEARCH NOT FOUND DELETE**

When a package is bound with a wild card (*) for package names, in the form of PKLIST(COL1.*,COL2.*.....), EDM generates a NOT-FOUND record to avoid future I/O if a collection ID/package name combination does not exist.

This field shows how often a record was removed from the cache during package binding.

**Field Name:** QISEKNFR

**STATEMENTS IN GLOBAL CACHE**

Number of statements in the global cache.

**Field Name:** QISESTMT
**Global DDF Activity**

This topic shows detailed information about “Statistics - Global DDF Activity”.

**Statistics - Global DDF Activity**

The field labels shown in the following sample layout of “Statistics - Global DDF Activity” are described in the following section.

<table>
<thead>
<tr>
<th>GLOBAL DDF ACTIVITY</th>
<th>QUANTITY /SECOND /THREAD /COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBAT/CONN QUEUED-MAX ACTIVE 0.00 0.00 0.00 N/A</td>
<td></td>
</tr>
<tr>
<td>CONN REJECTED-MAX CONNECTED 0.00 0.00 0.00 N/A</td>
<td></td>
</tr>
<tr>
<td>CONN CLOSED - MAX QUEUED 0.00 0.00 0.00 N/A</td>
<td></td>
</tr>
<tr>
<td>CONN CLOSED - MAX WAIT 0.00 0.00 0.00 N/A</td>
<td></td>
</tr>
<tr>
<td>COLD START CONNECTIONS 0.00 0.00 0.00 0.00</td>
<td></td>
</tr>
<tr>
<td>WARM START CONNECTIONS 0.00 0.00 0.00 0.00</td>
<td></td>
</tr>
<tr>
<td>RESYNCHRONIZATION ATTEMPTED 0.00 0.00 0.00 0.00</td>
<td></td>
</tr>
<tr>
<td>RESYNCHRONIZATION SUCCEEDED 0.00 0.00 0.00 0.00</td>
<td></td>
</tr>
<tr>
<td>CUR TYPE 1 INACTIVE DBATS 0.00 N/A N/A N/A</td>
<td></td>
</tr>
<tr>
<td>HWM TYPE 1 INACTIVE DBATS 2.00 N/A N/A N/A</td>
<td></td>
</tr>
<tr>
<td>TYPE 1 CONNECTIONS TERMINAT 0.00 0.00 N/A N/A</td>
<td></td>
</tr>
<tr>
<td>CUR INACTIVE CONNS (TYPE 2) 0.02 N/A N/A N/A</td>
<td></td>
</tr>
<tr>
<td>HWM INACTIVE CONNS (TYPE 2) 14.00 N/A N/A N/A</td>
<td></td>
</tr>
<tr>
<td>ACC QU INACT CONNS (TYPE 2) 2.00 0.00 N/A N/A</td>
<td></td>
</tr>
<tr>
<td>CUR QU INACT CONNS (TYPE 2) 0.00 N/A N/A N/A</td>
<td></td>
</tr>
<tr>
<td>MIN QUEUE TIME 0.000109 N/A N/A N/A</td>
<td></td>
</tr>
<tr>
<td>MAX QUEUE TIME 0.000109 N/A N/A N/A</td>
<td></td>
</tr>
<tr>
<td>AVG QUEUE TIME 0.000109 N/A N/A N/A</td>
<td></td>
</tr>
<tr>
<td>HWM QU INACT CONNS (TYPE 2) 8.00 N/A N/A N/A</td>
<td></td>
</tr>
<tr>
<td>CUR ACTIVE AND DISCON DBATS 0.00 N/A N/A N/A</td>
<td></td>
</tr>
<tr>
<td>HWM ACTIVE AND DISCON DBATS 11.00 N/A N/A N/A</td>
<td></td>
</tr>
<tr>
<td>HWM TOTL REMOTE CONNECTIONS 14.00 N/A N/A N/A</td>
<td></td>
</tr>
<tr>
<td>CUR DISCON DBATS NOT IN USE 0.00 N/A N/A N/A</td>
<td></td>
</tr>
<tr>
<td>HWM DISCON DBATS NOT IN USE 11.00 N/A N/A N/A</td>
<td></td>
</tr>
<tr>
<td>DBATS CREATED 1.00 N/A N/A N/A</td>
<td></td>
</tr>
<tr>
<td>DISCON (POOL) DBATS REUSED 1.00 N/A N/A N/A</td>
<td></td>
</tr>
<tr>
<td>CUR ACTIVE DBATS-BND DEALLC 0.00 N/A N/A N/A</td>
<td></td>
</tr>
<tr>
<td>HWM ACTIVE DBATS-BND DEALLC 0.00 N/A N/A N/A</td>
<td></td>
</tr>
</tbody>
</table>

**DBAT/CONN QUEUED-MAX ACTIVE**

The number of times a DBAT or connection was queued because it reached the ZPARM maximum for active remote threads (MAXDBAT).

**Field Name:** QDSTQDBT

This is an exception field.

**CONN REJECTED-MAX CONNECTED**

The number of connections that were rejected because the ZPARM limit for maximum remote connections (CONDBAT) was reached.

**Field Name:** QDSTQCRT

**CONN CLOSED - MAX QUEUED**

The number of queued client connections whose TCP/IP sockets were closed because the system parameter MAXCONQN was exceeded.
The socket close only occurs when the DB2 subsystem is a member of a data sharing group and DB2 was started with DDF THREADS set to INACTIVE.

**Field Name: QDSTNCQC**

**CONN CLOSED - MAX WAIT**

The number of queued client connections whose TCP/IP socket were closed due to system parameter MAXCONQW being exceeded.

The socket close only occurs when the DB2 subsystem is a member of a data sharing group and DB2 was started with DDF THREADS set to INACTIVE.

**Field Name: QDSTNCCW**

**COLD START CONNECTIONS**

The number of cold start connections with all remote locations (two-phase commit operations only).

**Field Name: QDSTCSTR**

This is an exception field.

**WARM START CONNECTIONS**

The number of warm start connections with all remote locations (two-phase commit operations only).

**Field Name: QDSTWSTR**

This is an exception field.

**RESYNCHRONIZATION ATTEMPTED**

The number of resynchronization connections attempted with all remote locations (two-phase commit operations only).

**Background and Tuning Information**

A large value can indicate network or system problems.

**Field Name: QDSTRSAT**

This is an exception field.

**RESYNCHRONIZATION SUCCEEDED**

The number of resynchronization connections that succeeded with all remote locations (two-phase commit operations only).

**Background and Tuning Information**

If the value of this field is much less than the number of resynchronizations attempted, network problems might exist.

**Field Name: QDSTRSSU**

This is an exception field.

**CUR TYPE 1 INACTIVE DBATS**

The current number of inactive DBATs type 1 (snapshot).

**Field Name: QDSTQCIT**

**HWM TYPE 1 INACTIVE DBATS**

The maximum number of inactive type 1 DBATs.
Global DDF Activity

This value is a high-water mark.

Field Name: QDSTQMIT
This is an exception field.

TYPE 1 CONNECTIONS TERMINAT
The number of threads or connections that were terminated instead of being made type 1 inactive because the maximum number of type 1 inactive threads was reached (MAXTYPE1).

Field Name: QDSTNITC

CUR INACTIVE CONNS (TYPE 2)
The current number of type 2 inactive connections.

Field Name: QDSTCIN2

HWM INACTIVE CONNS (TYPE 2)
The maximum number of concurrent type 2 inactive connections that existed.
This value is a high-water mark for QDSTCIN2.

Field Name: QDSTMIN2

ACC QU INACT CONNS (TYPE 2)
The number of RECEIVE requests on type 2 inactive or new connections that are queued to be serviced by a disconnected (pooled) DBAT.

Field Name: QDSTQIN2

CUR QU INACT CONNS (TYPE 2)
The current number of type 2 inactive or new connections that are queued waiting for a database access thread (DBAT).

Field Name: QDSTNQR2

MIN QUEUE TIME
The minimum queue time of a type 2 inactive or new connection that was queued waiting for a database access thread (DBAT) in the last statistical period.

Field Name: QDSTNQMN

MAX QUEUE TIME
The maximum queue time of a type 2 inactive or new connection that was queued waiting for a database access thread (DBAT) in the last statistical period.

Field Name: QDSTNQMX

AVG QUEUE TIME
The average queue time of a type 2 inactive or new connection that was queued waiting for a database access thread (DBAT) in the last statistical period.

Field Name: QDSTNQAV

HWM QU INACT CONNS (TYPE 2)
Global DDF Activity

The maximum number of type 2 inactive or new connections that are queued waiting for a database access thread.
This value is a high-water mark for QDSTNQR2.
Field Name: QDSTMQR2

CUR ACTIVE AND DISCON DBATS
The current number of active and disconnected (pooled) DBATs.
Field Name: QDSTCNAT

HWM ACTIVE AND DISCON DBATS
The maximum number of active and disconnected (pooled) DBATs that existed.
This value is a high-water mark for QDSTCNAT.
Field Name: QDSTHWAT
This is an exception field.

HWM TOTL REMOTE CONNECTIONS
The maximum number of active and remote connections. This value is a high-water mark.
Field Name: QDSTHWDT
This is an exception field.

CUR DISCON DBATS NOT IN USE
The current number of disconnected (pooled) DBATs that are available to process type 2 inactive or new connections.
Field Name: QDSTNADS

HWM DISCON DBATS NOT IN USE
The maximum number of disconnected (pooled) DBATs that are available to process type 2 inactive or new connections.
This value is a high-water mark for QDSTNADS.
Field Name: QDSTMADS

DBATS CREATED
The number of requests that required a database access thread (DBAT) to be created to process the request.

Note: This does not include database access threads created to replace disconnected (pooled) DBATs that terminated because they reached their reuse limit.
Field Name: QDSTNDBA

DISCON (POOL) DBATS REUSED
The number of requests that were satisfied by assigning a disconnected (pooled) DBAT to process the request.
Field Name: QDSTPOOL

CUR ACTIVE DBATS-BND DEALLC
Global DDF Activity

The current number of DBATs that are active because the associated packages were bound with RELEASE(DEALLOCATE).

Field Name: QDSTNARD

HWM ACTIVE DBATS-BND DEALLC

The maximum number of DBATs that are active because the associated packages were bound with RELEASE(DEALLOCATE).

Field Name: QDSTMARD
This topic shows detailed information about “Statistics - Group Buffer Pool Activity”.

This block shows activity for the group buffer pool connected to the reported DB2 system. The counters are cumulative from the time when the buffer pool was first connected. If more than one 4 KB or 32 KB group buffer pool block is printed, blocks showing the 4 KB and 32 KB group buffer pool totals are printed. If the report contains both 4 KB and 32 KB group buffer pool blocks, a block showing the totals of all group buffer pools is printed.

Statistics - Group Buffer Pool Activity

The field labels shown in the following sample layout of “Statistics - Group Buffer Pool Activity” are described in the following section.

```
GROUP BP0 QUANTITY /SECOND /THREAD /COMMIT
----------------------------- -------- ------- ------- -------
GROUP BP R/W RATIO (%)        71.22 N/A  N/A  N/A
GBP SYN.READ(XI) HIT RATIO(%) 5.51 N/A  N/A  N/A
GBP-DEPENDENT GETPAGES        6212.00 0.15 22.43 1.79
SYN.READ(XI)-DATA RETURNED    197.00 0.00 0.71 0.10
SYN.READ(XI)-NO DATA RETURN   3379.00 0.08 12.20 1.79
SYN.READ(NF)-DATA RETURNED    1.00 0.00 0.00 0.00
SYN.READ(NF)-NO DATA RETURN   0.00 0.00 0.00 0.00
UNREGISTER PAGE               0.00 0.00 0.00 0.00

CLEAN PAGES SYNC.WRITTEN      0.00 0.00 0.00 0.00
CLEAN PAGES ASYNC.WRITN       0.00 0.00 0.00 0.00
REG.PAGE LIST (RPL) REQUEST   115.00 0.00 0.42 0.06
NUMBER OF PAGES RETR.FROM GBP 0.00 0.00 0.00 0.00
PAGES CASTOUT                 37.00 0.00 0.13 0.02
UNLOCK CASTOUT                37.00 0.00 0.13 0.02

READ CASTOUT CLASS            223.00 0.01 0.81 0.12
READ DIRECTORY INFO           0.00 0.00 0.00 0.00
READ STORAGE STATISTICS       4907.00 0.12 17.71 2.59
REGISTER PAGE                 0.00 0.00 0.00 0.00
DELETE NAME                   93.00 0.00 0.34 0.06
ASYNCH GBP REQUESTS           2959.00 0.07 10.68 1.56
EXPlicit X-INVALIDATIONS      0.00 0.00 0.00 0.00
CASTOUT CLASS THRESHOLD       0.00 0.00 0.00 0.00
GROUP BP CASTOUT THRESHOLD    0.00 0.00 0.00 0.00
GBP CHECKPOINTS TRIGGERED    175.00 0.00 0.63 0.09
WRITE FAILED-NO STORAGE       0.00 0.00 0.00 0.00

WRITE TO SEC-GBP FAILED       0.00 0.00 0.00 0.00
COMPL CHECKS SUSPENDED        0.00 0.00 0.00 0.00
DELETE NAME LIST SEC-GBP      0.00 0.00 0.00 0.00
DELETE NAME FROM SEC-GBP      0.00 0.00 0.00 0.00
UNLOCK CASTOUT STATS SEC-GBP  0.00 0.00 0.00 0.00
ASYNCH SEC-GBP REQUESTS       0.00 0.00 0.00 0.00

GROUP BP0 CONTINUED QUANTITY /SECOND /THREAD /COMMIT
--------------------------------------------------- ------- ------- ------- -------
WRITE AND REGISTER            114.00 0.00 0.41 0.06
WRITE AND REGISTER MULT       24.00 0.00 0.09 0.01
CHANGED PGS SYNC.WRITN        278.00 0.01 1.00 0.15
CHANGED PGS ASYNC.WRITN       0.00 0.00 0.00 0.00
PAGES WRITE & REG MULT        164.00 0.00 0.59 0.09
READ FOR CASTOUT             37.00 0.00 0.13 0.02
READ FOR CASTOUT MULT         0.00 0.00 0.00 0.00
```
### Group Buffer Pool Activity

<table>
<thead>
<tr>
<th>Field</th>
<th>Value 1</th>
<th>Value 2</th>
<th>Value 3</th>
<th>Value 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>PAGE P-LOCK LOCK REQ</td>
<td>394.00</td>
<td>0.01</td>
<td>1.42</td>
<td>0.21</td>
</tr>
<tr>
<td>SPACE MAP PAGES</td>
<td>94.00</td>
<td>0.00</td>
<td>0.34</td>
<td>0.05</td>
</tr>
<tr>
<td>DATA PAGES</td>
<td>231.00</td>
<td>0.01</td>
<td>0.83</td>
<td>0.12</td>
</tr>
<tr>
<td>INDEX LEAF PAGES</td>
<td>69.00</td>
<td>0.00</td>
<td>0.25</td>
<td>0.04</td>
</tr>
<tr>
<td>PAGE P-LOCK UNLOCK REQ</td>
<td>440.00</td>
<td>0.01</td>
<td>1.59</td>
<td>0.23</td>
</tr>
<tr>
<td>PAGE P-LOCK LOCK SUSP</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>SPACE MAP PAGES</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DATA PAGES</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>INDEX LEAF PAGES</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>PAGES IN WRITE-AROUND</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>

#### GROUP BP R/W RATIO (%)

The group buffer pool read/write ratio. This reflects the effectiveness of
the group buffer pool and whether the GBPCACHE NONE option can be
used.

Field Name: SGBRWRAT

#### GBP SYN.READ(XI) HIT RATIO(%)

The percentage of all requests made to read a page from the group buffer
pool because the page was invalidated in the member's buffer pool, which
found the data in the group buffer pool and did not have to retrieve the
page from DASD.

Background and Tuning Information

For highly active group buffer pools, consider increasing the GBP size if
the SYN.READ(XI) HIT RATIO percent is smaller than 90.

Field Name: SGBXIRAT

#### GBP-DEPENDENT GETPAGES

The number of Getpages made for GBP-dependent objects.

Field Name: QBGLGG

#### SYN.READ(XI)-DATA RETURNED

The number of requests made to read a page from the group buffer pool
because the page was invalidated in the member's buffer pool. The
member found the required page in the group buffer pool.

Background and Tuning Information

When you increase the size of the group buffer pool (GBP), the number of
pages returned from the GBP can increase. Conversely, decreasing the size
of the GBP can cause DB2 to return fewer pages because the GBP cannot
hold pages long enough to allow them to be retrieved again.

Field Name: QBGLXD

This is an *exception* field.

#### SYN.READ(XI)-NO DATA RETURN

This is an additional field.
Group Buffer Pool Activity

The number of requests to read a page from the group buffer pool that were required because the page was invalidated in the member's buffer pool. The member did not find the data in the group buffer pool and had to retrieve the page from DASD.

Background and Tuning Information

Normally, when the page in a member's buffer is cross invalidated, the buffer is refreshed from the group buffer pool. In this instance, the requested page was not found in the group buffer pool though the page set is still GBP-dependent. The page has been removed from the group buffer pool for one of two reasons:

- Shortage of data pages and consequent reclamation of this page
- Shortage of directory entries and consequent removal of the page together with cross invalidation of that page in the local buffer pools of all members using that page.

If the value in this field is high, you may want to tune the group buffer pool (GBP). Depending on the reason, increase the number of GBP data pages, increase the size of the directory entry space, or increase both the number of GBP data pages and the space for directory entries. Oversizing the group buffer pool can cause unnecessary GBP checkpoint overhead.

Field Name: QBGLX3R

SYN.READ(NF)-DATA RETURNED

The number of requests made to read a page from the group buffer pool because the page was not in the buffer pool of the member. The member found the page in the group buffer pool.

Background and Tuning Information

The requesting member needs a page from a table space or index that is GBP-dependent or has GBPCACHE ALL defined. To get that page, the group buffer pool is checked before the page set on DASD.

If the group buffer pool is used to cache both clean and changed pages (GBPCACHE ALL is used for all data), you can try to get more pages returned from the group buffer pool by increasing the size of the group buffer pool. Do not tune the GBP based on this counter if it is used for caching changed pages only (GBPCACHE CHANGED).

Field Name: QBGLMD

This is an exception field.

SYN.READ(NF)-NO DATA RETURN

The number of requests made to read a page from the group buffer pool because the page was not in the member's buffer pool. The member did not find the required data in the group buffer pool and had to retrieve the page from DASD.

Background and Tuning Information

The requesting member needs a page from a table space or index that is GBP-dependent or has GBPCACHE ALL defined. To get that page, the group buffer pool is checked before the page set on DASD.

You can compare the value in this counter with the number of pages that were returned from the group buffer pool, see Sync.Read (Not Found) - Data Returned. If the group buffer pool is used to cache both clean and
Group Buffer Pool Activity

changed pages (GBPCACHE ALL is used for all data), you can try to get
more pages returned from the group buffer pool by increasing the size of
the group buffer pool. Do not tune the GBP based on this counter if it is
used for caching changed pages only (GBPCACHE CHANGED).

Field Name: QBGLMR

UNREGISTER PAGE
The number of times DB2 unregistered interest for a single page. This
happens when DB2 steals pages from the member's buffer pool that belong
to GBP-dependent page sets or partitions.

Background and Tuning Information
A large value here indicates that the local buffer pool contains a mixture of
GBP-dependent data and non-GBP-dependent data.

The page stolen from the local buffer pool is replaced by a new one. This
counter makes a distinction on whether the new page depends on the
group buffer pool or not.

Usually a page of a GBP-dependent page set or partition is replaced by a
page that is also GBP-dependent. In this instance, the unregister request for
the page being stolen is combined with the read and register request for
the new page. These combined requests do not contribute to this counter.

If, however, a page of a GBP-dependent page set or partition is replaced by
a page that is not GBP-dependent, then only an unregister request is sent
to the coupling facility. These separate requests are counted here.

Field Name: QBGLDG

CLEAN PAGES SYNC.WRTN
The number of clean pages that were synchronously written to the group
buffer pool from the virtual pool.

Background and Tuning Information
Only GBPCACHE ALL causes clean (unchanged) pages to be written to the
coupling facility. The pages are written to the coupling facility even if the
page set is not GBP-dependent. If group buffer pool caching works
effectively for prefetch, the value in this field should be much smaller than
the value in Synchronous Read (Not Found) - Data Returned.

Field Name: QBGLWC

This is an exception field.

CLEAN PAGES ASYNC.WRTN
The number of clean pages that were asynchronously written to the group
buffer pool from the virtual pool.

Background and Tuning Information
Only GBPCACHE ALL causes clean (unchanged) pages to be written to the
group coupling facility. In this instance pages are written even if the page
set is not GBP-dependent. Asynchronous write is done under prefetch
processing.

If group buffer pool caching works effectively for prefetch, the value in this
field should be much smaller than the combined values in:
• Synchronous Read (Not Found) - Data Returned
Group Buffer Pool Activity

- Asynchronous Reads - Data Returned
- Clean pages - Read after register page list

**Field Name:** QBGLAC
This is an exception field.

**REG.PAGE LIST (RPL) REQUEST**
The number of register page list (RPL) requests made by prefetch. The group buffer pool must be allocated in a group coupling facility with CFLEVEL=2 or higher.

**Background and Tuning Information**
Performance might be improved by enabling RPL.

**Field Name:** QBGLAX
This is an exception field.

**NUMBER OF PAGES RETR.FROM GBP**
The number of coupling facility reads performed by prefetch to retrieve a changed page from the group buffer pool.

**Field Name:** QBGLAY
This is an exception field.

**PAGES CASTOUT**
The number of data pages that were cast out of the group buffer pool of the member.

Castout to a page set or partition is done by the castout owner of the page set or partition. This is normally the DB2 subsystem that had the first update intent on the page set or partition.

**Background and Tuning Information**
The number of pages written per I/O is normally close to the value of this field divided by the value in Unlock Castout.

For example, if an average of four pages is written per castout write I/O, the number of pages cast out should be four times the number in this field.

Because DB2 usually includes more than one page in the request to write pages to DASD, the number in this field should always be significantly more than Unlock Castout. If it is not (for example, when "unlock castout" is more than half of "pages castout"), the castout write I/O is inefficient; probably because you have random update patterns on the DB2 data or a low castout threshold.

**Field Name:** QBGLRC
This is an exception field.

**UNLOCK CASTOUT**
The number of times DB2 issued an unlock request to the coupling facility for completed castout I/Os.

When pages are cast out to DASD, they are locked for castout in the coupling facility. This castout lock is not an IRLM lock; it is to ensure that only one system can cast out a given page at a time.

**Background and Tuning Information**
The number of pages written per I/O is normally close to the value of pages castout divided by the value of this field.

For example, if an average of four pages is written per castout write I/O, the number of pages cast out should be four times the value in this field.

Because DB2 usually includes more than one page in a write request, the number in this field should always be significantly less than pages castout. If it is not (for example, when "unlock castout" is more than half of "pages castout"), the castout write I/O is inefficient; possibly because you have random update patterns on the DB2 data or a low castout threshold.

**Field Name:** QBGLUN

**READ CASTOUT CLASS**

The number of requests made to the group buffer pool to determine which pages, from a particular page set or partition, must be cast out because they are cached as changed pages.

This request is issued either by the page set or partition castout owner, or, when the group buffer pool castout threshold is reached, by the group buffer pool structure owner.

**Field Name:** QBGLCC

**READ DIRECTORY INFO**

The number of requests issued by the group buffer pool structure owner to read the directory entries of all changed pages in the group buffer pool.

This request is issued at group buffer pool checkpoints to record the oldest recovery log record sequence number (LRSN). It is used as a basis for recovery if the group buffer pool fails.

Such requests might have to be issued several times for each group buffer pool checkpoint to read the directory entries for all changed pages.

**Background and Tuning Information**

If the value of this counter appears to be abnormally high, consider upgrading the coupling facility to CFLEVEL=2 or higher to raise the number of directory entries that can be read with one request. You can also increase the group buffer pool checkpoint interval, but this can lengthen the recovery for the group buffer pool.

**Field Name:** QBGLRD

**READ STORAGE STATISTICS**

The number of times DB2 requested statistics information from the group buffer pool. It is issued by the group buffer pool structure owner at timed intervals to determine whether the group buffer pool castout threshold (GBPOOLT) has been reached.

**Field Name:** QBGLOS

**REGISTER PAGE**

The number of times DB2 registered interest in a single page.

These are "register-only" requests, which means that DB2 is not requesting any data back from the request.
This request is made only to create a directory entry for the page to be used for cross-invalidation when the page set or partition P-lock is downgraded from S to IS mode, or from SIX to IX mode.

Field Name: QBGLRG

DELETE NAME

The number of requests made by DB2 to delete directory and data entries associated with a particular page set or partition from the group buffer pool.

DB2 issues this request when it changes a page set or partition from GBP-dependent to non GBP-dependent. DB2 also issues this request for objects that are defined with GBPCACHE ALL when those objects are first opened.

Background and Tuning Information

This counter is a measure of how often page sets or partitions change between being and not being dependent on the group buffer pool.

You can prevent DB2 going in and out of GBP dependency too often by tuning the following subsystem parameters that affect data sets when they are switched to a different state:

PCLOSEN
Pseudoclose frequency. The number of checkpoints required before a data set that was not updated can be a pseudoclose candidate.

If the PCLOSEN condition is met, the page set or partition is converted from read-write to read-only state. Depending on other concurrent users, this could raise the chance for the page set or partition to go out of GBP dependency.

PCLOSET
Pseudoclose time. The amount of time (in minutes) that must elapse before a data set can be a pseudoclose candidate.

If the PCLOSEN or PCLOSET condition is met, the page set or partition is converted from read-write to read-only state. Depending on other concurrent users, this could raise the chance for the page set or partition to go out of GBP dependency.

LOGLOAD
The number of log records that DB2 writes between successive checkpoints.

These parameters are specified in the CHECKPOINT FREQ field in panel DSNTIPN.

Field Name: QBGLDN

ASYNCH GBP REQUESTS

The number of IXLCACHE invocations for the primary group buffer pool.

Field Name: QBGLHS

EXPLICIT X-INVALIDATIONS

The number of times an explicit coupling facility cross-invalidation request was issued.

Field Name: QBGLEX
Group Buffer Pool Activity

**CASTOUT CLASS THRESHOLD**

The number of times group buffer pool castout was initiated because the group buffer pool class castout threshold was detected.

**Background and Tuning Information**

The class castout threshold is one of two group buffer pool thresholds. In most cases the default value for the class threshold (10 percent) is a good choice. Depending on your workload, altering this value can reduce DASD contention during castout.

*Field Name: QBGLCT*

This is an *exception* field.

**GROUP BP CASTOUT THRESHOLD**

The number of times a group buffer pool castout was initiated because the group buffer pool castout threshold was detected.

**Background and Tuning Information**

The GBP castout threshold, GBP class castout threshold, and the length of the GBP checkpoint interval determine the castout characteristics of the group buffer pool.

You can consider this threshold a safety margin to protect the group buffer pool from being accidentally flooded by overactive applications.

In most situations, the default value for the group buffer pool castout threshold of 50 percent is a good choice. Use the ALTER GROUPBUFFERPOOL command to tune the group buffer pool thresholds.

*Field Name: QBGLGT*

This is an *exception* field.

**GBP CHECKPOINTS TRIGGERED**

The number of group buffer pool checkpoints triggered by this member.

**Background and Tuning Information**

The value of this counter depends on the length of the group buffer pool checkpoint interval.

*Field Name: QBGLCK*

**WRITE FAILED-NO STORAGE**

The number of coupling facility write requests that could not complete due to a lack of coupling facility storage resources.

**Background and Tuning Information**

A value greater than zero indicates that the data page resources of the coupling facility are being consumed faster than the DB2 castout processes can free them.

On write failure, the affected DB2 member initiates castout and retries several times, and finally, if it is a changed page, it will be added to the logical page list (LPL) requiring recovery.

If the problem is not simply due to a momentary surge in activity, you need either to decrease the group buffer pool castout thresholds, or to increase the number of data entries in the group buffer pool. To increase the number of data entries, you can do one of the following:
Group Buffer Pool Activity

- Increase the total size of the group buffer pool.
- Adjust the ratio of directory entries to data entries in favor of data entries.

**Field Name**: QBGLWF

This is an *exception* field.

**WRITE TO SEC-GBP FAILED**

The number of coupling facility requests to write changed pages to the secondary group buffer pool for duplexing that failed because of a lack of storage in the coupling facility.

**Field Name**: QBGL2F

This is an *exception* field.

**COMPL CHECKS SUSPENDED**

The number of completion checks for writes to the secondary GBP that were suspended because the write had not yet been completed.

**Field Name**: QBGL2S

**DELETE NAME LIST SEC-GBP**

The number of DELETE NAME LIST requests to delete pages from the secondary group buffer pool that have just been cast out from the primary.

**Field Name**: QBGL2D

**DELETE NAME FROM SEC-GBP**

The number of group buffer pool requests to delete a page from the secondary group buffer pool. These requests are issued by the group buffer pool structure owner to delete orphaned data entries in the secondary GBP as part of the garbage collection logic.

**Field Name**: QBGL2N

**UNLOCK CASTOUT STATS SEC-GBP**

The number of coupling facility requests to read the castout statistics for the secondary group buffer pool. These requests are issued by the group buffer pool structure owner to check for orphaned data entries in the secondary group buffer pool.

**Field Name**: QBGL2R

**ASYNCH SEC-GBP REQUESTS**

The number of asynchronous IXLCACHE invocations for the secondary group buffer pool.

**Field Name**: QBGL2H

**WRITE AND REGISTER**

The number of Write and Register requests.

**Field Name**: QBGLWS

**WRITE AND REGISTER MULT**

The number of Write and Register Multiple requests.

**Field Name**: QBGLWM

**CHANGED PGS SYNC.WRTN**
**Group Buffer Pool Activity**

The number of changed pages written synchronously to the group buffer pool.

Pages are written with Write and Register (WAR) requests or Write and Register Multiple (WARM) requests.

At commit time changed pages are forced from the virtual buffer pool of the member to the coupling facility.

**Background and Tuning Information**

In data sharing, changed pages must have been written to the group buffer pool by the time a transaction commits. The pages are written either synchronously (force at commit) or asynchronously, for example, when a local buffer pool threshold is reached or at a member's checkpoint. The number of pages that have to be forced out synchronously (in “burst mode”) at commit time can be reduced if asynchronous writes are triggered more frequently.

You can use the vertical deferred write threshold (VDWQT) to reduce the number of pages that have to be forced out synchronously and to increase the number of pages that are asynchronously written before the transaction commits. For GBP-dependent page sets, writes triggered by the vertical deferred write threshold go to the coupling facility. You can cause changed pages to be written out quicker and in smaller increments, by reducing the vertical deferred write threshold (VDWQT).

**Field Name:** QBGLSW

This is an exception field.

**CHANGED PGS ASYNC.WRTN**

The number of changed pages written asynchronously to the group buffer pool.

Pages are written in response to Write and Register (WAR) and Write and Register Multiple (WARM) requests.

Changed pages can be written from the member’s virtual buffer pool to the group coupling facility before the application commits. This happens when, for example, a local buffer pool threshold is reached, or when P-lock negotiation forces the pages on the vertical deferred write queue to be written to the group buffer pool.

**Background and Tuning Information**

In data sharing, changed pages must have been written to the group buffer pool before a transaction commits. The pages are written either synchronously during commit processing or asynchronously before the transaction commits when, for example, a local buffer pool threshold is reached or at a member's checkpoint. See Changed Pages - Written Synchronously for the number of changed pages synchronously written to the group buffer pool.

The vertical deferred write threshold (VDWQT) can be used to reduce the number of pages that have to be forced out synchronously and to increase the number of pages that are asynchronously written before the transaction commits. For GBP-dependent page sets, writes triggered by the vertical deferred write threshold go to the coupling facility. If you want changed pages to be written out quicker and in smaller increments, you can lower the vertical deferred write threshold (VDWQT).
Group Buffer Pool Activity

Field Name: QBGLAW
This is an exception field.

PAGES WRITE & REG MULT
The number of pages written using Write and Register Multiple (WARM) requests.
Field Name: QBGLWP

READ FOR CASTOUT
The number of Read For Castout requests. One page read per request.
Field Name: QBGLCR

READ FOR CASTOUT MULT
The number of Read For Castout Multiple requests.
Field Name: QBGLCM

PAGE P-LOCK LOCK REQ
The sum of all page P-lock lock requests.
Field Name: SBGLPLR

SPACE MAP PAGES
The number of page P-lock lock requests for space map pages.
Field Name: QBGLP1

DATA PAGES
The number of page P-lock lock requests for data pages.
Field Name: QBGLP2

INDEX LEAF PAGES
The number of page P-lock lock requests for index leaf pages.
Field Name: QBGLP3

PAGE P-LOCK UNLOCK REQ
The number of page P-lock unlock requests.
Field Name: QBGLU1

PAGE P-LOCK LOCK SUSP
The sum of all page P-lock lock suspensions.
Field Name: SBGLPLS

SPACE MAP PAGES
The number of page P-lock lock suspensions for space map pages.
Field Name: SBGLPLS

DATA PAGES
The number of page P-lock lock suspensions for data pages.
Field Name: QBGLS1

INDEX LEAF PAGES
The number of page P-lock lock suspensions for index leaf pages.
Group Buffer Pool Activity

Field Name: QBGLS3

PAGE P-LOCK LOCK NEG
The sum of all page P-lock lock negotiations.

Field Name: SBGLPLN

SPACE MAP PAGES
The number of page P-lock lock negotiations for space map pages.

Field Name: QBGLN1

DATA PAGES
The number of page P-lock lock negotiations for data pages.

Field Name: QBGLN2

INDEX LEAF PAGES
The number of page P-lock lock negotiations for index leaf pages.

Field Name: QBGLN3

PAGES IN WRITE-AROUND
The number of pages written to DASD directly using the write-around protocol.

Field Name: QBGLWA
This topic shows detailed information about “Statistics - Highlights”.

The sample shows the Statistics Highlights block for the long report. The description also shows additional fields printed with the Statistics short report.

**Statistics - Highlights**

The field labels shown in the following sample layout of “Statistics - Highlights” are described in the following section.

--- HIGHLIGHTS -----------------------------------------------------------------------------------------------
INTERVAL START : 07/26/10 19:32:45.57 SAMPLING START: 07/26/10 19:32:45.57 TOTAL THREADS : 0.00
INTERVAL END : 07/26/10 19:35:47.34 SAMPLING END : 07/26/10 19:35:47.34 TOTAL COMMITS : 33.00
INTERVAL ELAPSED: 3:01.768843 OUTAGE ELAPSED: 0.000000 DATA SHARING MEMBER: N/A

**INTERVAL START**

The start time of the period represented by this report or trace entry.

For a trace, it is the timestamp of the DB2 Statistics records pair which marks the beginning of the delta record represented by the trace entry.

For the group page of group-scope reports it is the beginning of the earliest interval across reported members.

For SAVE data, it is the timestamp of the first DB2 Statistics report pair used to derive a row in the statistics SAVE and FILE tables.

For FILE, it is the timestamp of the DB2 Statistics records pair which marks the beginning of the delta record represented by a row in the statistics SAVE and FILE tables.

**Field Name:** SDBEGREC

**INTERVAL END**

The end time of the period represented by this report or trace entry.

For a trace, it is the timestamp of the DB2 statistics records pair which marks the end of the delta record represented by the trace entry.

For the group page of group-scope reports it is the ending of the latest interval across reported members.

For SAVE data, it is the timestamp of the last DB2 statistics report pair used to derive a row in the statistics SAVE and FILE tables.

For FILE, it is the timestamp of the DB2 statistics records pair which marks the end of the delta record represented by a row in the statistics SAVE and FILE tables.

**Field Name:** SDENDREC

**INTERVAL ELAPSED**

The elapsed time of the period represented by this report or trace entry.

For a trace, it is the time elapsed between two consecutive DB2 statistics records pairs which mark the beginning and the end of the delta record represented by the trace entry. For a report, it is the elapsed time for the period within the interval record for which the DB2 statistics data is available.

For the group page of group-scope reports it is the average elapsed time of all the reported members.
Field Name: SDELTIME

SAMPLING START

The timestamp of the first DB2 statistics records pair used to derive a report entry (an interval record). For example, when INTERVAL(0) is specified, the sampling start coincides with the interval record start time in member-scope reports.

Field Name: SDSAMPST

SAMPLING END

The timestamp of the last DB2 statistics records pair used to derive a report entry (an interval record). For example, when INTERVAL(0) is specified, the sampling start coincides with the interval record end time in member-scope reports.

Field Name: SDSAMPEN

OUTAGE ELAPSED

The time for which OMEGAMON XE for DB2 PE detected discontinuity in the available DB2 statistics data. The most common reason for this is a stop or start of the reported DB2 system within the reported interval. For the group page of group-scope reports it is the average outage time of all reported members.

Field Name: SDOUTEL

TOTAL THREADS

The number of successful create thread requests. It does not include DBATs.

A thread is required before an application can use SQL. When established, a thread can have one or more secondary authorization IDs.

A thread is needed to perform any DB2 activity. For example, a thread is needed to run a DB2 utility to perform an IFI request such as READS, or to process a DB2 command such as -DISPLAY THREAD. However, a thread is not created if the command failed because of a syntax error.

Background and Tuning Information

Thread reuse can help improve performance.

The term "thread reuse" only applies to IMS and CICS attachments. In the case of the TSO attachment facility and the call attachment facility (CAF), threads cannot be reused, because the threads are allocated to the user address space.

Thread reuse should be considered in the following cases:

• If transaction volume is high:

  High volume transactions should achieve a high percentage of thread reuse. If threads are reused on low volume transactions, the number of threads needed increases because these threads are not automatically terminated by IMS when not being used. This may result in too many idle threads for the level of the DB2 workload. Under CICS, protected threads are terminated after about 45 seconds if no transaction eligible to reuse the thread has been received.

• If thread creation cost is significant:
As a rule of thumb, more than 5% of the total CPU cost of transaction processing is considered significant.

The ACQUIRE and RELEASE parameters of BIND should be specified to minimize the thread creation cost, while providing the needed concurrency:

- If most of the application plan's SQL statements are executed, then ACQUIRE(ALLOCATE) is cheaper than ACQUIRE(USE).
- If only a small number of the SQL statements are executed, ACQUIRE(USE) becomes cheaper and improves concurrency, because the required resources are only acquired (locked) when the plan actually references (uses) them. An example would be a generalized plan used by many different transactions. It would contain multiple logic paths referencing different tables.

Note that, when packages are involved, ACQUIRE(USE) is always implicitly used.

- Concurrency in thread reuse is based on page locking provided by the IS and IX intent locks, whose duration is governed by ACQUIRE and RELEASE of BIND.

RELEASE(DEALLOCATE) is strongly recommended for thread-reuse transactions to reduce transaction CPU time.

When thread reuse is implemented, monitor the EDM pool. It should be sufficient in size to accommodate expanding plans where the next transaction requires additional plan sections over those that are already part of the plan.

Field Name: Q3STCTHD
This is an exception field.

TOTAL COMMITS
The total number of commits during the interval covered by the report or trace. This includes commit, read-only commit, sync, and rollback events. DBATs executed on this location are not included.

Field Name: SDCOMMIT
This is an exception field.

DATA SHARING MEMBER
In group-scope reports, this field shows the name of the member for which statistics is presented, and, on the group total page, the number of DB2 subsystems in the reported data sharing group. In member-scope reports, this field shows N/A.

Field Name: QWHAMEMN
This is an exception field.

INCREMENTAL BINDS
The number of incremental binds (excluding prepare). It is incremented by:

- SQL statements with BIND VALIDATE(RUN) that fail at bind time and are bound again at execution time
- Static DDL statements (such as CREATE TABLE, DROP TABLE, LOCK TABLE) that use DB2 private protocol

Highlights
If a plan is bound with VALIDATE(RUN), DB2 performs validity checks at bind time and rechecks any failures at run time. This can result in catalog contention and degraded application performance, depending on the number of statements flagged and how many times they are executed. Avoid VALIDATE(RUN) if possible. Ensure that all objects are created and all privileges are granted before bind, and select the VALIDATE(BIND) option.

**Field Name:** QXINCRB
This is an *exception* field.

**DBAT QUEUED**
The number of times a DBAT or connection was queued because it reached the ZPARM maximum for active remote threads (MAXDBAT).

**Field Name:** QDSTQDBT
This is an *exception* field.

**AUTH SUCC.W/OUT CATALOG**
The number of successful authorization checks that do not use the DB2 catalog (including plan cache checks and public checks).

**Background and Tuning Information**
For transaction level security, ENABLE and DISABLE on BIND PACKAGE should be used to ensure adequate security. Granting execute authority on the plan to public should be adequate.

**Field Name:** QTAUCCH

**DB2 Command**
The total number of DB2 commands that were issued.

**Field Name:** SDSTTOTL

**BUFUPDT/PAGES WRITTEN**
The number of buffer updates per page written from the buffer pool to DASD.

The ratio of BUFFER UPDATES (QBSTWS) to PAGES WRITTEN (QBSTPWS) suggests a high level of efficiency as the ratio increases, because more updates are being externalized per physical write. For example, if there are 10 updates on the same page before it is externalized, then the ratio is 10:1 or 10. If all 10 updates are on 10 distinct pages, then the ratio is 10:10 or 1.

**Background and Tuning Information**
Buffer updates per pages written depends strongly on the type of application. For example, a batch program that processes a table in skip sequential mode with a high row update frequency in a dedicated environment can achieve very good update efficiency. In contrast, update efficiency tends to be lower for transaction processing applications, because transaction processing tends to be random.

The following factors can influence the number of updates per page:

**Number of rows per page**
A small PCTFREE value will gather more rows on the same page. However, at the same time this can have impact on concurrency.
Buffer pool size and deferred write thresholds
Increase DWQT and VDWQT or the size of the buffer pool. This would tell DB2 to let page updates accumulate in the buffer pool. This means, the probability that more updates per page get captured increases. This effect is less significant if the buffer pool is concurrently used by multiple transactions, it depends on the type of transaction.

Field Name: SBRBUPW

TOTAL API
The total number of calls made to IFI.
Field Name: SDIFITOT

PAGES WRITTEN/WRITE I/O
The number of pages written from the buffer pool to DASD per synchronous or asynchronous write I/O. This count does not include preformatting I/O, such as I/O needed to prepare a data set for use.

Background and Tuning Information
The following factors impact the ratio of pages written per write I/O:

Checkpoint frequency
At checkpoint time, I/Os are scheduled to write all updated pages on the deferred write queue to DASD. If this occurs too frequently, the deferred write queue does not grow large enough to achieve a high ratio of pages written per write I/O.

The checkpoint frequency depends on the number of logs written between two consecutive checkpoints. This number is set at installation time; see the field CHECKPOINT FREQ of installation panel DSNTPIN.

Frequency of active log switch
DB2 takes a system checkpoint each time the active log is switched. High frequency of active log switches causes the problem described under checkpoint frequency.

Buffer pool size and deferred write thresholds
The deferred write thresholds (VDWQT and DWQT) are a function of buffer pool size. If the buffer pool size is decreased, these thresholds are reached more frequently, causing I/Os to be scheduled more often to write some of the pages on the deferred write queue to DASD. This prevents the deferred write queue from growing large enough to achieve a high ratio of pages written per write I/O.

Number of data sets, and the spread of updated pages across them
The efficiency of write I/O also depends on the number of data sets associated with the buffer pool and spread of updated pages across them. Because of the nature of batch processing, the ratio of pages written to write I/Os can be expected to be higher than that expected for transaction type workloads.

To determine update efficiency check also the ratio Buffer Updates / Pages Written (SBRBUPW).

Field Name: SBRPWWIO
MEMBER

In group-scope reports, this field shows the name of the member for which statistics is presented, and, on the group total page, the number of DB2 subsystems in the reported data sharing group. In member-scope reports, this field shows N/A.

Field Name: QWHAMEMN
Statistics - IFC Destinations

The field labels shown in the following sample layout of “Statistics - IFC Destinations” are described in the following section.

<table>
<thead>
<tr>
<th>IFC DEST.</th>
<th>WRITTEN</th>
<th>NOT WRITN</th>
<th>BUF.OVER</th>
<th>NOT ACCEP</th>
<th>WRT.FAIL</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMF</td>
<td>40.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>GTF</td>
<td>0.00</td>
<td>0.00</td>
<td>N/A</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>OP1</td>
<td>0.00</td>
<td>0.00</td>
<td>N/A</td>
<td>0.00</td>
<td>N/A</td>
</tr>
<tr>
<td>OP2</td>
<td>0.00</td>
<td>0.00</td>
<td>N/A</td>
<td>0.00</td>
<td>N/A</td>
</tr>
<tr>
<td>OP3</td>
<td>0.00</td>
<td>0.00</td>
<td>N/A</td>
<td>0.00</td>
<td>N/A</td>
</tr>
<tr>
<td>OP4</td>
<td>0.00</td>
<td>0.00</td>
<td>N/A</td>
<td>0.00</td>
<td>N/A</td>
</tr>
<tr>
<td>OP5</td>
<td>0.00</td>
<td>0.00</td>
<td>N/A</td>
<td>0.00</td>
<td>N/A</td>
</tr>
<tr>
<td>OP6</td>
<td>0.00</td>
<td>0.00</td>
<td>N/A</td>
<td>0.00</td>
<td>N/A</td>
</tr>
<tr>
<td>OP7</td>
<td>0.00</td>
<td>0.00</td>
<td>N/A</td>
<td>0.00</td>
<td>N/A</td>
</tr>
<tr>
<td>OP8</td>
<td>0.00</td>
<td>0.00</td>
<td>N/A</td>
<td>0.00</td>
<td>N/A</td>
</tr>
<tr>
<td>RES</td>
<td>0.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>TOTAL</td>
<td>40.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td></td>
</tr>
</tbody>
</table>

SMF - WRITTEN

The total number of SMF records successfully written.

Field Name: SDISMFWR

GTF - WRITTEN

The total number of GTF records successfully written.

Field Name: SDIGTFWR

OP1 - WRITTEN

The total number of OP1 records successfully written.

Field Name: SDIOP1WR

OP2 - WRITTEN

The total number of OP2 records successfully written.

Field Name: SDIOP2WR

OP3 - WRITTEN

The total number of OP3 records successfully written.

Field Name: SDIOP3WR

OP4 - WRITTEN

The total number of OP4 records successfully written.

Field Name: SDIOP4WR

OP5 - WRITTEN

The total number of OP5 records successfully written.

Field Name: SDIOP5WR

OP6 - WRITTEN

The total number of OP6 records successfully written.
IFC Destinations

Field Name: SDIOP6WR
OP7 - WRITTEN
  The total number of OP7 records successfully written.
  Field Name: SDIOP7WR

Field Name: SDIOP8WR
OP8 - WRITTEN
  The total number of OP8 records successfully written.
  Field Name: SDIOP8WR

Field Name: SDIIRTWR
RES - WRITTEN
  The total number of RES records successfully written.
  Field Name: SDIIRTWR

Field Name: SDITOTW
TOTAL WRITTEN
  The total number of IFC records successfully written.
  Field Name: SDITOTW

Field Name: SDISMFMNW
SMF - NOT WRITTEN
  The total number of SMF records not written. This field should be 0.
  Otherwise, records may have been lost.
  Field Name: SDISMFMNW

Field Name: SDIGTFNW
GTF - NOT WRITTEN
  The total number of GTF records not written. This field should be 0.
  Otherwise, records may have been lost.
  Field Name: SDIGTFNW

Field Name: SDIOP1NW
OP1 - NOT WRITTEN
  The total number of OP1 records not written. This field should be 0.
  Otherwise, records may have been lost.
  Field Name: SDIOP1NW

Field Name: SDIOP2NW
OP2 - NOT WRITTEN
  The total number of OP2 records not written. This field should be 0.
  Otherwise, records may have been lost.
  Field Name: SDIOP2NW

Field Name: SDIOP3NW
OP3 - NOT WRITTEN
  The total number of OP3 records not written. This field should be 0.
  Otherwise, records may have been lost.
  Field Name: SDIOP3NW

Field Name: SDIOP4NW
OP4 - NOT WRITTEN
  The total number of OP4 records not written. This field should be 0.
  Otherwise, records may have been lost.
  Field Name: SDIOP4NW

Field Name: SDIOP5NW
OP5 - NOT WRITTEN
  The total number of OP5 records not written. This field should be 0.
  Otherwise, records may have been lost.
Field Name: SDIOP5NW

**OP6 - NOT WRITTEN**

The total number of OP6 records not written. This field should be 0. Otherwise, records may have been lost.

Field Name: SDIOP6NW

**OP7 - NOT WRITTEN**

The total number of OP7 records not written. This field should be 0. Otherwise, records may have been lost.

Field Name: SDIOP7NW

**OP8 - NOT WRITTEN**

The total number of OP8 records not written. This field should be 0. Otherwise, records may have been lost.

Field Name: SDIOP8NW

**TOTAL NOT WRITTEN**

The total number of IFC records not written.

Field Name: SDTOTNW

**SMF - BUF.OVER**

The total number of SMF buffer overruns. Ideally, this field should be 0 or very small.

Field Name: SDISMFBF

**SMF - NOT ACCP**

The total number of SMF records not accepted. Ideally, this field should be 0 or very small.

Field Name: SDISMFRA

**GTF - NOT ACCP**

The total number of GTF records not accepted. Ideally, this field should be 0 or very small.

Field Name: SDIGTFRA

**OP1 - NOT ACCP**

The total number of OP1 records not accepted. Ideally, this field should be 0 or very small.

Field Name: SDIOP1RA

**OP2 - NOT ACCP**

The total number of OP2 records not accepted. Ideally, this field should be 0 or very small.

Field Name: SDIOP2RA

**OP3 - NOT ACCP**

The total number of OP3 records not accepted. Ideally, this field should be 0 or very small.

Field Name: SDIOP3RA
IFC Destinations

**OP4 - NOT ACCP**

The total number of OP4 records not accepted. Ideally, this field should be 0 or very small.

**Field Name:** SDIOP4RA

**OP5 - NOT ACCP**

The total number of OP5 records not accepted. Ideally, this field should be 0 or very small.

**Field Name:** SDIOP5RA

**OP6 - NOT ACCP**

The total number of OP6 records not accepted. Ideally, this field should be 0 or very small.

**Field Name:** SDIOP6RA

**OP7 - NOT ACCP**

The total number of OP7 records not accepted. Ideally, this field should be 0 or very small.

**Field Name:** SDIOP7RA

**OP8 - NOT ACCP**

The total number of OP8 records not accepted. Ideally, this field should be 0 or very small.

**Field Name:** SDIOP8RA

**TOTAL NOT ACCP**

The total number of IFC records not accepted.

**Field Name:** SDTOTNA

**SMF - WRT.FAIL**

The total number of SMF write failures. Ideally, this field should be 0 or very small.

**Field Name:** SDISMFWF

**GTF - WRT.FAIL**

The total number of GTF write failures. Ideally, this field should be 0 or very small.

**Field Name:** SDIGTWF

**TOTAL WRT.FAIL**

The total number of IFC write failures.

**Field Name:** SDTOTWF
### IFC Record Counts

This topic shows detailed information about “Statistics - IFC Record Counts”.

#### Statistics - IFC Record Counts

The field labels shown in the following sample layout of “Statistics - IFC Record Counts” are described in the following section.

<table>
<thead>
<tr>
<th>Field Label</th>
<th>Written</th>
<th>Not Written</th>
</tr>
</thead>
<tbody>
<tr>
<td>IFC RECORD COUNTS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SYSTEM RELATED</td>
<td>4.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DATABASE RELATED</td>
<td>4.00</td>
<td>0.00</td>
</tr>
<tr>
<td>ACCOUNTING</td>
<td>4.00</td>
<td>0.00</td>
</tr>
<tr>
<td>START TRACE</td>
<td>2.00</td>
<td>0.00</td>
</tr>
<tr>
<td>STOP TRACE</td>
<td>1.00</td>
<td>0.00</td>
</tr>
<tr>
<td>SYSTEM PARAMETERS</td>
<td>5.00</td>
<td>0.00</td>
</tr>
<tr>
<td>SYS.PARMS-BPOOLS</td>
<td>4.00</td>
<td>0.00</td>
</tr>
<tr>
<td>AUDIT</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td><strong>24.00</strong></td>
<td><strong>0.00</strong></td>
</tr>
</tbody>
</table>

**SYSTEM RELATED (WRITTEN)**
- The number of system-related records written.
  - **Field Name**: SDISRRW

**DATABASE RELATED (WRITTEN)**
- The number of database-related records written.
  - **Field Name**: SDIDRRW

**ACCOUNTING (WRITTEN)**
- The number of accounting records written.
  - **Field Name**: SDIACTW

**START TRACE (WRITTEN)**
- The number of start trace records written.
  - **Field Name**: SDISTRW

**STOP TRACE (WRITTEN)**
- The number of stop trace records written.
  - **Field Name**: SDISTPW

**SYSTEM PARAMETERS (WRITTEN)**
- The number of DB2 system parameter records written.
  - **Field Name**: SDIZPMW

**SYS.PARMS-BPOOLS (WRITTEN)**
- The number of DB2 system parameter buffer pool records written.
  - **Field Name**: SDBSCRSW

**AUDIT (WRITTEN)**
- The number of DB2 audit records written.
  - **Field Name**: SDIAUDW

**TOTAL (WRITTEN)**
IFC Record Counts

The total number of records that were successfully written.

Field Name: SDTSCRSW

SYSTEM RELATED (NOT WRTN)

The number of system-related records not written. Ideally, this field should be 0 or very small.

Field Name: SDISRRN

DATABASE RELATED (NOT WRTN)

The number of database-related records not written. Ideally, this field should be 0 or very small.

Field Name: SDIDRRN

ACCOUNTING (NOT WRTN)

The number of accounting records not written. Ideally, this field should be 0 or very small.

Field Name: SDIACTN

START TRACE (NOT WRTN)

The number of start trace records not written. Ideally, this field should be 0 or very small.

Field Name: SDISTRN

STOP TRACE (NOT WRTN)

The number of stop trace records not written. Ideally, this field should be 0 or very small.

Field Name: SDISTPN

SYSTEM PARAMETERS (NOT WRTN)

The number of DB2 system parameter records not written. Ideally, this field should be 0 or very small.

Field Name: SDIZPMN

SYS.PARMS-BPOOLS (NOT WRTN)

The number of DB2 system parameter buffer pool records not written. Ideally, this field should be 0 or very small.

Field Name: SDBSCRNW

AUDIT (NOT WRTN)

The number of DB2 audit records that were not written.

Field Name: SDIAUDN

TOTAL (NOT WRTN)

The total number of records that were not written.

Field Name: SDTSCRNW
This topic shows detailed information about “Statistics - IRLM Storage Below and Above 2 GB (DB2 11)”.

**Statistics - IRLM Storage Below and Above 2 GB (DB2 11)**

The field labels shown in the following sample layout of “Statistics - IRLM Storage Below and Above 2 GB (DB2 11)” are described in the following section.

### IRLM Storage Below and Above 2 GB

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXTENDED CSA SIZE IN USE (MB)</td>
<td>The total amount of Extended Common Service Area (ECSA) storage in use by Internal Resource Lock Manager (IRLM) pools (DB2 field: QW0225L_BBECSA).</td>
</tr>
<tr>
<td>HWM EXTENDED CSA SIZE IN USE (MB)</td>
<td>The high-water mark of ECSA storage allocated by IRLM pools (DB2 field: QW0225L_BBECSAH).</td>
</tr>
<tr>
<td>31 BIT PRIVATE IN USE (MB)</td>
<td>The total amount of 31-bit private storage in use by IRLM pools (DB2 field: QW0225L_BBPVT).</td>
</tr>
<tr>
<td>HWM 31 BIT PRIVATE IN USE (MB)</td>
<td>The high-water mark of 31-bit private storage allocated by IRLM pools (DB2 field: QW0225L_BBPVH).</td>
</tr>
<tr>
<td>THRESHOLD 31 BIT PRIVATE (MB)</td>
<td>The threshold of 31-bit private storage available for normal IRLM execution. Only requests for storage by “must complete” tasks will be granted if this threshold is exceeded (DB2 field: QW0225L_BPMAX).</td>
</tr>
<tr>
<td>64 BIT PRIVATE IN USE (MB)</td>
<td>The total amount of 64-bit private storage in use by IRLM pools (DB2 field: QW0225L_ABPVT).</td>
</tr>
<tr>
<td>HWM 64 BIT PRIVATE IN USE (MB)</td>
<td>The high-water mark of 64-bit private storage allocated by IRLM pools (DB2 field: QW0225L_ABPVH).</td>
</tr>
<tr>
<td>THRESHOLD 64 BIT PRIVATE (MB)</td>
<td>The threshold of 64-bit private storage available for normal IRLM execution. Only requests for storage by “must complete” tasks will be granted if this threshold is exceeded (DB2 field: QW0225L_ABMX).</td>
</tr>
<tr>
<td>64 BIT COMMON IN USE (MB)</td>
<td>The total amount of 64-bit common storage in use by IRLM pools (DB2 field: QW0225L_ABPVT).</td>
</tr>
<tr>
<td>HWM 64 BIT COMMON IN USE (MB)</td>
<td>The high-water mark of 64-bit common storage allocated by IRLM pools (DB2 field: QW0225L_ABPVH).</td>
</tr>
</tbody>
</table>
IRLM Storage Below and Above 2 GB

Field Name: S225IAPU

HWM 64 BIT PRIVATE IN USE (MB)

The high-water mark of 64-bit private storage allocated by IRLM pools
(DB2 field: QW0225I_ABPVH).

Field Name: S225IAPH

THRESHOLD 64 BIT PRIVATE (MB)

The threshold of 64-bit private storage available for normal IRLM
execution. Only requests for storage by "must complete" tasks will be
granted if this threshold is exceeded (DB2 field: QW0225I_APMAX).

Field Name: S225IAPT

64 BIT COMMON IN USE (MB)

The total amount of 64-bit common storage in use by IRLM pools (DB2
field: QW0225I_ABCSA).

Field Name: S225IACU

HWM 64 BIT COMMON IN USE (MB)

The high-water mark of 64-bit common storage allocated by IRLM pools
(DB2 field: QW0225I_ABCSH).

Field Name: S225IACH
Latch Counters

This topic shows detailed information about “Statistics - Latch Counters”.

The QVLS latch counters represent the number of suspends that were performed by agents that attempted to obtain a latch.

There is not a one-to-one relationship between the QVLS counters and IFCID 56 or 57, because an agent might suspend multiple times or not at all, while trying to obtain a latch. That is why the QVLS counters are not directly related to Accounting Class 3.

Statistics - Latch Counters

The field labels shown in the following sample layout of “Statistics - Latch Counters” are described in the following section.

<table>
<thead>
<tr>
<th>LATCH CNT</th>
<th>/SECOND</th>
<th>/SECOND</th>
<th>/SECOND</th>
<th>/SECOND</th>
</tr>
</thead>
<tbody>
<tr>
<td>LC01-LC04</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>LC05-LC08</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>LC09-LC12</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>LC13-LC16</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>LC17-LC20</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>LC21-LC24</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.01</td>
</tr>
<tr>
<td>LC25-LC28</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>LC29-LC32</td>
<td>0.00</td>
<td>0.01</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>LC254</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

LC01

This field is infrequently used.

Field Name: QVLSLC01

LC02

The predominant latch usage is: Global authorization cache.

Field Name: QVLSLC02

LC03

The predominant latch usage is: DDF disconnect.

Field Name: QVLSLC03

LC04

This field is infrequently used.

Field Name: QVLSLC04

LC05

The predominant latch usage is: IRLM data sharing exits or RLF.

Field Name: QVLSLC05

LC06

The predominant latch usage is: Data sharing index split.

Field Name: QVLSLC06

LC07

The predominant latch usage is: Index lotch and OBD allocation.
Latch Counters

Field Name: QVLSLC07
LC08
The predominant latch usage is: Query parallelism.
Field Name: QVLSLC08
LC09
The predominant latch usage is: Utilities or stored procedure URIDs.
Field Name: QVLSLC09
LC10
The predominant latch usage is: Allied agent chain or sequence descriptors.
Field Name: QVLSLC10
LC11
This field is infrequently used.
Field Name: QVLSLC11
LC12
The predominant latch usage is: Global transaction ID table.
Field Name: QVLSLC12
LC13
The predominant latch usage is: Pageset operations.
Field Name: QVLSLC13
LC14
The predominant latch usage is: Bufferpool LRU.
Field Name: QVLSLC14
LC15
The predominant latch usage is: ARCHIVE LOG MODE(QUIESCE).
Field Name: QVLSLC15
LC16
This field is infrequently used.
Field Name: QVLSLC16
LC17
The predominant latch usage is: RURE chain.
Field Name: QVLSLC17
LC18
The predominant latch usage is: DDF resynch list.
Field Name: QVLSLC18
LC19
The predominant latch usage is: Log write.
Field Name: QVLSLC19
Latch Counters

LC20
The predominant latch usage is: System checkpoint.
Field Name: QVLSLC20

LC21
The predominant latch usage is: Accounting rollup.
Field Name: QVLSLC21

LC22
The predominant latch usage is: Internal checkpoint.
Field Name: QVLSLC22

LC23
The predominant latch usage is: Buffer manager.
Field Name: QVLSLC23

LC24
The predominant latch usage is: EDM pool or prefetch.
Field Name: QVLSLC24

LC25
The predominant latch usage is: Workfile allocation.
Field Name: QVLSLC25

LC26
The predominant latch usage is: Dynamic statement cache.
Field Name: QVLSLC26

LC27
The predominant latch usage is: Stored procedures or authorization cache.
Field Name: QVLSLC27

LC28
The predominant latch usage is: Stored procedures or authorization cache.
Field Name: QVLSLC28

LC29
The predominant latch usage is: Field procs and DDF transaction manager.
Field Name: QVLSLC29

LC30
The predominant latch usage is: Agent services.
Field Name: QVLSLC30

LC31
The predominant latch usage is: Storage manager.
Field Name: QVLSLC31

LC32
Latch Counters

The predominant latch usage is: Storage manager.
Field Name: QVLSLC32

LC254
The predominant latch usage is: Index latch.
Field Name: QVLSLC254
### Statistics - Locking Activity

The field labels shown in the following sample layout of “Statistics - Locking Activity” are described in the following section.

<table>
<thead>
<tr>
<th>LOCKING ACTIVITY</th>
<th>QUANTITY</th>
<th>/SECOND</th>
<th>/THREAD</th>
<th>/COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUSPENSIONS (ALL)</td>
<td>15.00</td>
<td>0.08</td>
<td>N/C</td>
<td>0.45</td>
</tr>
<tr>
<td>SUSPENSIONS (LOCK ONLY)</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>SUSPENSIONS (IRLM LATCH)</td>
<td>15.00</td>
<td>0.08</td>
<td>N/C</td>
<td>0.45</td>
</tr>
<tr>
<td>SUSPENSIONS (OTHER)</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>TIMEOUTS</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DEADLOCKS</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>LOCK REQUESTS</td>
<td>11626.00</td>
<td>63.96</td>
<td>N/C</td>
<td>352.30</td>
</tr>
<tr>
<td>UNLOCK REQUESTS</td>
<td>2745.00</td>
<td>15.10</td>
<td>N/C</td>
<td>83.18</td>
</tr>
<tr>
<td>QUERY REQUESTS</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>CHANGE REQUESTS</td>
<td>389.00</td>
<td>2.14</td>
<td>N/C</td>
<td>11.79</td>
</tr>
<tr>
<td>OTHER REQUESTS</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>LOCK ESCALATION (SHARED)</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>LOCK ESCALATION (EXCLUSIVE)</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DRAIN REQUESTS</td>
<td>7.00</td>
<td>0.04</td>
<td>N/C</td>
<td>0.21</td>
</tr>
<tr>
<td>DRAIN REQUESTS FAILED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>CLAIM REQUESTS</td>
<td>999.00</td>
<td>5.50</td>
<td>N/C</td>
<td>30.27</td>
</tr>
<tr>
<td>CLAIM REQUESTS FAILED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
</tbody>
</table>

#### SUSPENSIONS (ALL)

The total number of suspensions.

**Field Name:** SLRSUSP

#### SUSPENSIONS (LOCK ONLY)

The number of times a lock could not be obtained and the unit of work was suspended.

**Background and Tuning Information**

This number should be low, ideally 0.

The number of lock suspensions is a function of the lock requests. Lock suspensions (or conflicts) can happen on either LOCK REQUEST or CHANGE REQUEST.

Suspensions are highly dependent on the application and table space locking protocols.

**Field Name:** QTXASLOC

This is an exception field.

#### SUSPENSIONS (IRLM LATCH)

The number of latch suspensions.

**Field Name:** QTXASLAT

This is an exception field.

#### SUSPENSIONS (OTHER)
Locking Activity

The number of suspensions caused by something other than lock or latch.

**Field Name:** QTXASOTH

This is an exception field.

**TIMEOUTS**

The number of times a unit of work was suspended for a time exceeding the timeout value. This number should be low, ideally 0.

**Field Name:** QTXATIM

This is an exception field.

**DEADLOCKS**

The number of times deadlocks were detected. This number should be low, ideally 0.

**Background and Tuning Information**

Deadlocks occur when two or more application processes each hold locks on resources that the others need, without which they cannot proceed. Ensure that all applications accessing the same tables access them in the same order.

Deadlocks can also occur through index page splits if there is high insert activity. In this case, the recommendation is to set SUBPAGES to 1 for the index.

This field is incremented once for each deadlock encountered. There is no correlation between this field and the deadlock events reported in the Locking report set or the number of IFCID 172 records written. This field reports all deadlocks, regardless of how they were resolved. The locking report and record trace IFCID 172 show only those deadlocks that were resolved by DB2.

**Field Name:** QTXADEA

This is an exception field.

**LOCK REQUESTS**

The number of requests to lock a resource.

**Field Name:** QTXALOCK

This is an exception field.

**UNLOCK REQUESTS**

The number of requests to unlock a resource.

This value can be less than the number of lock requests because DB2 can release several locks with a single unlock request.

**Field Name:** QTXAUNLK

**QUERY REQUESTS**

The number of query requests.

**Field Name:** QTXAQRY

**CHANGE REQUESTS**

The number of change requests.

**Field Name:** QTXACHG
OTHER REQUESTS
The number of requests to IRLM to perform a function other than LOCK, UNLOCK, QUERY, or CHANGE.

Field Name: QTXAIRLM

LOCK ESCALATION (SHARED)
The number of times the maximum page locks per table space are exceeded, and the table space lock escalates from a page lock (IS) to a table space lock (S) for this thread. You can specify the number of locks allowed per table space with the LOCKS PER TABLE(SPACE) parameter on the DB2 install panel DSNTIPJ.

Background and Tuning Information
Escalations can cause unpredictable response times. Lock escalations should only happen when an application process updates or references (if repeatable read is used) more pages than normal.

Field Name: QTXALES

This is an exception field.

LOCK ESCALATION (EXCLUSIVE)
The number of times the maximum page locks per table space are exceeded and the table space lock escalates from a page lock (IX) to a table space lock (X).

Background and Tuning Information
Escalations can cause unpredictable response times. Lock escalations should only happen when an application process updates or references (if repeatable read is used) more pages than it normally does.

A useful rule of thumb is to compare the number of escalations (shared and exclusive) to the successful escalations (those that did not cause deadlocks and timeouts). If this value, or the number Lock escalations - shared and if the number of timeouts or deadlocks is also not 0, the timeout or deadlock is probably caused by the escalation.

If many escalations cause deadlocks and timeouts, the recommendation is to change the escalation threshold value. Use of ANY is extremely useful to prevent unnecessary and expensive page locks, for example locking all pages in a tablespace.

Lock escalations, shared or exclusive, should not be expected in a transaction environment.

Field Name: QTXALEX

This is an exception field.

DRAIN REQUESTS
The number of drain requests.

Field Name: QTXADRNO

This is an exception field.

DRAIN REQUESTS FAILED
The number of unsuccessful drain requests.

Field Name: QTXADRUN
Locking Activity

This is an exception field.

CLAIM REQUESTS

The number of claim requests.

Field Name: QTXACLNO

This is an exception field.

CLAIM REQUESTS FAILED

The number of unsuccessful claim requests.

Field Name: QTXACLUN

This is an exception field.
This topic shows detailed information about “Statistics - Log Activity”.

### Statistics - Log Activity

The field labels shown in the following sample layout of “Statistics - Log Activity” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Quantity</th>
<th>/Second</th>
<th>/Thread</th>
<th>/Commit</th>
</tr>
</thead>
<tbody>
<tr>
<td>READS SATISFIED-OUTPUT BUFF</td>
<td>96.00</td>
<td>0.53</td>
<td>N/C</td>
<td>2.91</td>
</tr>
<tr>
<td>READS SATISFIED-OUTP.BUF(%)</td>
<td>100.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>READS SATISFIED-ACTIVE LOG</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>READS SATISFIED-ACTV.LOG(%)</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>READS SATISFIED-ARCHIVE LOG</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>READS SATISFIED-ARCH.LOG(%)</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TAPE VOLUME CONTENTION WAIT</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>READ DELAYED-UNAVAIL.RESOUR</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>ARCHIVE LOG READ ALLOCATION</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>ARCHIVE LOG WRITE ALLOCAT.</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>CONTR.INTERV.OFFLOADED-ARCH</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>LOOK-AHEAD MOUNT ATTEMPTED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>LOOK-AHEAD MOUNT SUCCESSFUL</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>UNAVAILABLE OUTPUT LOG BUFF</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>OUTPUT LOG BUFFER PAGED IN</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>LOG RECORDS CREATED</td>
<td>19276.00</td>
<td>106.05</td>
<td>N/C</td>
<td>584.12</td>
</tr>
<tr>
<td>LOG CI CREATED</td>
<td>861.00</td>
<td>4.74</td>
<td>N/C</td>
<td>26.09</td>
</tr>
<tr>
<td>LOG WRITE I/O REQ (LOG1&amp;2)</td>
<td>554.00</td>
<td>3.05</td>
<td>N/C</td>
<td>16.79</td>
</tr>
<tr>
<td>LOG CI WRITTEN (LOG1&amp;2)</td>
<td>2004.00</td>
<td>11.02</td>
<td>N/C</td>
<td>60.73</td>
</tr>
<tr>
<td>LOG RATE FOR 1 LOG (MB)</td>
<td>N/A</td>
<td>0.02</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>

**READS SATISFIED-OUTPUT BUFF**

The number of log reads satisfied from the output buffer.

**Background and Tuning Information**

This field, together with the reads satisfied from active log and reads satisfied from archive log (QJSTRACT and QJSTRARH) fields indicate how efficiently DB2 retrieves log records. Use these numbers to adjust the number of output buffers and the total active log capacity to maximize DB2 performance.

**Field Name:** QJSTRBUF

This is an exception field.

**READS SATISFIED-OUTP.BUF(%)**

The percentage of log reads that were satisfied in the output log buffer.

**Field Name:** SARLRBUF

**READS SATISFIED-ACTIVE LOG**

The number of log reads satisfied from the active log data set.

**Background and Tuning Information**

This field, together with the reads satisfied from archive log and reads satisfied from output buffer fields, indicate how efficiently DB2 retrieves
Log Activity

log records. Use these numbers to adjust the number of output buffers and the total active log capacity to maximize DB2 performance. Ideally, this value should be 0 or very small.

Field Name: QJSTRACT
This is an exception field.

READS SATISFIED-ACTV.LOG(%)  
The percentage of log reads satisfied from the active log.

Field Name: SARLRACT
This is an exception field.

READS SATISFIED-ARCHIVE LOG  
The number of log reads satisfied from the archive log data set.

Background and Tuning Information
This field, together with the reads satisfied from active log and reads satisfied from output buffer fields indicate how efficiently DB2 retrieves log records. Use these numbers to adjust the number of output buffers and the total active log capacity to maximize DB2 performance. Ideally, this value should be 0 or very small.

Field Name: QJSTRARH
This is an exception field.

READS SATISFIED-ARCH.LOG(%)  
The percentage of log reads that were satisfied from the archive log data set.

Field Name: SARLRARC
This is an exception field.

TAPE VOLUME CONTENTION WAIT  
The number of read accesses that were delayed because of a tape volume contention when only one reader per tape is possible.

Background and Tuning Information
This field shows the number of agents forced to wait because a tape volume was already in use by another. If this number is not 0, increase the read tape units on the archive log data set parameters panel DSNTIPA.

Field Name: QJSTTVC
This is an exception field.

READ DELAYED-UNAVAIL.RESOUR  
The number of read accesses delayed due to unavailable resources.

Background and Tuning Information
Generally, this can be due to insufficient tape units allocated. If this is so, reissue the SET ARCHIVE command and use a higher value for the count parameter. Another (although unlikely) cause is insufficient archive log read service task availability.

Field Name: QJSTWUR
This is an exception field.
ARCHIVE LOG READ ALLOCATION
The number of archive log read allocations.
It indicates the frequency of archive log open and close activity.

Background and Tuning Information
A high number indicates a need for more or larger active log data sets.
This value should be small, ideally 0.

Field Name: QJSTALR
This is an exception field.

ARCHIVE LOG WRITE ALLOCATIONS
The number of archive log write allocations.
It indicates the frequency of archive log open and close activity.

Background and Tuning Information
A high number indicates a need for more or larger active log data sets.
This value should be small, ideally 0.

Field Name: QJSTALW

CONTR.INTERV.OFFLOADED-ARCH
The number of control intervals (CIs) offloaded from the active log to the archive log.

Field Name: QJSTCIOF

LOOK-AHEAD MOUNT ATTEMPTED
The number of look ahead (tape volume) mounts attempted.

Background and Tuning Information
This field and field QJSTLAMS (label LOOK-AHEAD MOUNT SUCCESSFUL) show the efficiency of look ahead for tape mounts.

Field Name: QJSTLAMA

LOOK-AHEAD MOUNT SUCCESSFUL
The number of successful look-ahead (tape volume) mounts. It indicates the look-ahead mounting performance gains.

Background and Tuning Information
For maximum performance, this field and field QJSTLAMA (label LOOK-AHEAD MOUNT ATTEMPTED) should be equal. To find the number of failed attempts, subtract the value in this field from LOOK-AHEAD MOUNT ATTEMPTED. Too many failed attempts negate potential performance gains. This can be caused by not having enough tape units available. Issue the DISPLAY ARCHIVE command and note the current count value. Then issue the SET ARCHIVE command using a higher value for the count parameter.

Field Name: QJSTLAMS

UNAVAILABLE OUTPUT LOG BUFF
The number of waits caused by an unavailable output log buffer.

When DB2 wants to write a log record and the log buffer is not available, DB2 and the application must wait for an available log buffer.
Log Activity

**Background and Tuning Information**

Another possible cause is that the size of the write threshold might be too close to the size of the output buffer.

If this field is not 0, increase the number in the output buffer field on installation panel DSNTIPL to increase the number of output buffers or increase the size of the buffer.

**Field Name:** QJSTWTB

This is an exception field.

**OUTPUT LOG BUFFER PAGED IN**

The number of times an output log buffer had to be paged in before it could be initialized. The log-write latch is held at this point.

**Background and Tuning Information**

A nonzero value could indicate that the output log buffer size is too large, or there is insufficient real storage to back up the output log buffer size.

**Field Name:** QJSTBPAG

**LOG RECORDS CREATED**

The number of log write requests.

The log record is written asynchronously to the log buffer. The application does not wait for the record to be written to the log data set and regains control immediately.

Buffered log records are written to DASD when the buffer threshold is exceeded.

**Field Name:** QJSTWRNW

**LOG CI CREATED**

The number of active log output control intervals created.

**Background and Tuning Information**

Log records are placed sequentially in output log buffers, which are formatted as VSAM control intervals. The control intervals are written to a set of predefined DASD active log data sets, which are used sequentially and recycled.

The ratio of this field to write output log buffers should be low.

**Rules of thumb:**

The lower the value, the better. A high value indicates that too many I/Os are required for the number of log buffers created.

It is possible that WRTTHTSRH is set too low. It is also possible that transactions could be arriving so infrequently that at commit time force requests are not queued and each force request is individually triggering an I/O of its log buffers.

**Field Name:** QJSTBFFL

**LOG WRITE I/O REQ (LOG1&2)**

The total number of log-write I/O requests (such as media manager calls).

This is the sum of the IFCID 038/039 pairs and includes both copy1 and copy2 active log data set writes.
Background and Tuning Information

This value should correspond to the active log write I/O activity in an RMF report.

Field Name: QJSTLOGW

LOG CI WRITTEN (LOG1&2)

The total number of log control intervals (CIs) written. This includes CI rewrites and both copy1 and copy2 active log data set writes. If a given CI is rewritten 5 times, this counter is incremented by 5.

Field Name: QJSTCIWR

LOG RATE FOR 1 LOG (MB)

The log rate for the active log data sets in MB per second. This figure is valid for dual logging, if single logging is used, multiply the value shown by 2.

Background and Tuning Information

To calculate this rate (mega bytes/second) at which data is written to the active log data set, multiply the value of field QJSTCIWR (label LOG CI WRITTEN (LOG1&2)) by 4096 and divide it by 1024 * 1024 * statistics-interval-seconds * 2. When the value exceeds 10MB/sec per log copy, you should examine I/O tuning of log data sets (for example, using faster log devices and/or I/O striping, using variable-length or compressed log record layouts to reduce log data size).

Field Name: SJSTCIWR
This topic shows detailed information about “Statistics - Miscellaneous”.

**Statistics - Miscellaneous**

The field labels shown in the following sample layout of “Statistics - Miscellaneous” are described in the following section.

<table>
<thead>
<tr>
<th>MISCELLANEOUS VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>HIGH LOG RBA 00000000001698125C05</td>
</tr>
<tr>
<td>BYPASS COL 0.00</td>
</tr>
<tr>
<td>MAX SQL CASCADING LEVEL 0.00</td>
</tr>
<tr>
<td>MAX STOR LOB VALUES (MB) 0.00</td>
</tr>
<tr>
<td>MAX STOR XML VALUES (MB) 0.00</td>
</tr>
<tr>
<td>ARRAY EXPANSIONS 0.00</td>
</tr>
<tr>
<td>SPARSE IX DISABLED 0.00</td>
</tr>
<tr>
<td>SPARSE IX BUILT WF 0.00</td>
</tr>
</tbody>
</table>

**HIGH LOG RBA**

The high-used RBA address of the log (DB2 field prior to DB2 11: QWSDLR).

*Field Name:* QWSDLRG

**BYPASS COL**

The total number of columns (rows x columns) for which an invalid select procedure was encountered.

DB2 bypasses invalid select procedures which can cause some degradation in performance.

*Field Name:* QISTCOLS

**MAX SQL CASCAD LEVEL**

The maximum level of indirect SQL cascading. This includes cascading because of triggers, UDFs, or stored procedures.

*Field Name:* QXCASCDP

This is an exception field.

**MAX STOR LOB VALUES (MB)**

Maximum storage used for LOB values.

*Field Name:* QXSTLOBV

This is an exception field.

**MAX STOR XML VALUES (MB)**

Maximum storage used for XML values.

*Field Name:* QXSTXMLV

**ARRAY EXPANSIONS**

The number of times an array variable is expanded beyond 32 KB.

*Field Name:* QXSTARRAY_EXPANSIONS

**SPARSE IX DISABLED**

The number of times that sparse index was disabled because of insufficient storage.
Field Name: QXSISTOR

SPARSE IX BUILT WF

The number of times that sparse-index built a physical work file for probing.

Field Name: QXSIWF
MVS LPAR Shared Storage Above 2 GB

This topic shows detailed information about “Statistics - MVS LPAR Shared Storage Above 2 GB (DB2 10 or later)”.

Statistics - MVS LPAR Shared Storage Above 2 GB (DB2 10 or later)

The field labels shown in the following sample layout of “Statistics - MVS LPAR Shared Storage Above 2 GB (DB2 10 or later)” are described in the following section.

<table>
<thead>
<tr>
<th>MVS LPAR SHARED STORAGE ABOVE 2 GB</th>
<th>QUANTITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>SHARED MEMORY OBJECTS</td>
<td>2.00</td>
</tr>
<tr>
<td>64 BIT SHARED STORAGE (MB)</td>
<td>163840.00</td>
</tr>
<tr>
<td>HWM FOR 64 BIT SHARED STORAGE (MB)</td>
<td>491520.00</td>
</tr>
<tr>
<td>64 BIT SHARED STORAGE BACKED IN REAL (MB)</td>
<td>2240.80</td>
</tr>
<tr>
<td>AUX STORAGE USED FOR 64 BIT SHARED (MB)</td>
<td>0.00</td>
</tr>
<tr>
<td>64 BIT SHARED STORAGE PAGED IN FROM AUX (MB)</td>
<td>0.00</td>
</tr>
<tr>
<td>64 BIT SHARED STORAGE PAGED OUT TO AUX (MB)</td>
<td>0.00</td>
</tr>
</tbody>
</table>

SHARED MEMORY OBJECTS

The number of shared memory objects allocated for this MVS LPAR (DB2 field: QW0225SHRMOMB).

**Field Name:** SW225SMO

64 BIT SHARED STORAGE (MB)

The amount of 64-bit shared storage allocated for this MVS LPAR (including hidden pages).

**Field Name:** S225SPG

HWM FOR 64 BIT SHARED STORAGE (MB)

High water mark of 64-bit shared storage allocated for this MVS LPAR (DB2 field: QW0225SHRGBYTES).

**Field Name:** SW225SGB

64 BIT SHARED STORAGE BACKED IN REAL (MB)

The amount of 64-bit shared storage backed in real storage for this MVS LPAR.

**Field Name:** S225SRL

AUX STORAGE USED FOR 64 BIT SHARED (MB)

The amount of auxiliary storage used for 64-bit shared storage for this MVS LPAR (including reserved auxiliary slots for pages that are paged in).

**Field Name:** S225SAX

64 BIT SHARED STORAGE PAGED IN FROM AUX (MB)

The amount of 64-bit shared storage paged in from auxiliary storage for this MVS LPAR.

**Field Name:** S225SPI

64 BIT SHARED STORAGE PAGED OUT TO AUX (MB)

The amount of 64-bit shared storage paged out to auxiliary storage for this MVS LPAR.
Field Name: S225SPO
Open/Close Activity

This topic shows detailed information about “Statistics - Open/Close Activity”.

Statistics - Open/Close Activity

The field labels shown in the following sample layout of “Statistics - Open/Close Activity” are described in the following section.

<table>
<thead>
<tr>
<th>OPEN/CLOSE ACTIVITY</th>
<th>QUANTITY</th>
<th>/SECOND</th>
<th>/THREAD</th>
<th>/COMMITH</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPEN DATASETS - HWM</td>
<td>79.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>OPEN DATASETS</td>
<td>64.34</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>DS NOT IN USE, NOT CLOSE-HWM</td>
<td>79.00</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>DS NOT IN USE, NOT CLOSED</td>
<td>55.43</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>IN USE DATA SETS</td>
<td>8.91</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>DSETS CLOSED-THRESH.REACHED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DSETS CONVERTED R/W -&gt; R/O</td>
<td>7.00</td>
<td>0.04</td>
<td>N/C</td>
<td>0.21</td>
</tr>
</tbody>
</table>

OPEN DATASETS - HWM

The maximum number of data sets concurrently open since the last time DB2 was started. This is a high-water mark (HWM).

Background and Tuning Information

Monitor this field to see whether you are reaching the maximum number of open data sets permissible. The maximum number currently is 10000.

Field Name: QTMAXDS

This is an exception field.

OPEN DATASETS

The number of data sets concurrently open (snapshot).

Field Name: QTDSOPN

DS NOT IN USE, NOT CLOSE-HWM

The maximum number of data sets on the deferred close queue. It is a high-water mark representing the maximum number of data sets that are not in use but have not been physically closed yet.

Field Name: QTMAXPB

This is an exception field.

DS NOT IN USE, NOT CLOSED

The number of data sets that are not currently used, but are not closed due to a deferred close (snapshot).

Field Name: QTSLWDD

This is an exception field.

IN USE DATA SETS

The number of data sets currently in use (snapshot).

Field Name: SDINUSEC

This is an exception field.

DSETS CLOSED-THRESH.REACHED
Open/Close Activity

The number of data sets that were closed because the total number of open data sets reached the deferred close threshold value. The deferred close value is based on the value of DSMAX or the MVS DD limit (whichever is smaller).

Field Name: QTDSDRN

This is an exception field.

DSETS CONVERTED R/W -> R/O

The number of infrequently updated data sets that are converted from R/W to R/O state. An updated data set is considered infrequently updated when it has not been updated for either 5 consecutive DB2 checkpoints or 60 minutes. For tablespace data sets, the switching from R/W to R/O state means the SYSLGRNG entry is closed.

Field Name: QTPCCT

This is an exception field.
Plan/Package Activity

This topic shows detailed information about “Statistics - Plan/Package Activity”.

Statistics - Plan/Package Activity

The field labels shown in the following sample layout of “Statistics - Plan/Package Activity” are described in the following section.

<table>
<thead>
<tr>
<th>PLAN/PACKAGE PROCESSING</th>
<th>QUANTITY</th>
<th>/SECOND</th>
<th>/THREAD</th>
<th>/COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>INCREMENTAL BINDS</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>PLAN ALLOCATION ATTEMPTS</td>
<td>2.00</td>
<td>0.01</td>
<td>N/C</td>
<td>0.06</td>
</tr>
<tr>
<td>PLAN ALLOCATION SUCCESSFUL</td>
<td>2.00</td>
<td>0.01</td>
<td>N/C</td>
<td>0.06</td>
</tr>
<tr>
<td>PACKAGE ALLOCATION ATTEMPT</td>
<td>17.00</td>
<td>0.09</td>
<td>N/C</td>
<td>0.52</td>
</tr>
<tr>
<td>PACKAGE ALLOCATION SUCCESS</td>
<td>17.00</td>
<td>0.09</td>
<td>N/C</td>
<td>0.52</td>
</tr>
<tr>
<td>PLANS BOUND</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>BIND ADD SUBCOMMANDS</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>BIND REPLACE SUBCOMMANDS</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>TEST BINDS NO PLAN-ID</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>PACKAGES BOUND</td>
<td>14.00</td>
<td>0.08</td>
<td>N/C</td>
<td>0.42</td>
</tr>
<tr>
<td>BIND ADD PACKAGE SUBCOMMAND</td>
<td>49.00</td>
<td>0.27</td>
<td>N/C</td>
<td>1.48</td>
</tr>
<tr>
<td>BIND REPLACE PACKAGE SUBCOMMAND</td>
<td>14.00</td>
<td>0.08</td>
<td>N/C</td>
<td>0.42</td>
</tr>
<tr>
<td>AUTOMATIC BIND ATTEMPTS</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>AUTOMATIC BINDS SUCCESSFUL</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>AUTO.BIND INVALID RES. IDS</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>AUTO.BIND PACKAGE ATTEMPTS</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>AUTO.BIND PACKAGES SUCCESS</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>REBIND SUBCOMMANDS</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>ATTEMPTS TO REBIND A PLAN</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>PLANS REBOUND</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>REBIND PACKAGE SUBCOMMANDS</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>ATTEMPTS TO REBIND PACKAGE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>PACKAGES REBOUND</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>FREE PLAN SUBCOMMANDS</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>ATTEMPTS TO FREE A PLAN</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>PLANS FREED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>FREE PACKAGE SUBCOMMANDS</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>ATTEMPTS TO FREE A PACKAGE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>PACKAGES FREED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
</tbody>
</table>

INCREMENTAL BINDS

The number of incremental binds (excluding prepare). It is incremented by:

- SQL statements with BIND VALIDATE(RUN) that fail at bind time and are bound again at execution time
- Static DDL statements (such as CREATE TABLE, DROP TABLE, LOCK TABLE) that use DB2 private protocol

Background and Tuning Information

If a plan is bound with VALIDATE(RUN), DB2 performs validity checks at bind time and rechecks any failures at run time. This can result in catalog contention and degraded application performance, depending on the number of statements flagged and how many times they are executed. Avoid VALIDATE(RUN) if possible. Ensure that all objects are created and all privileges are granted before bind, and select the VALIDATE(BIND) option.

Field Name: QXINCRB
Plan/Packagge Activity

This is an exception field.

PLAN ALLOCATION ATTEMPTS

The number of times a request was made to allocate a bound plan for an agent.

It represents the number of times DB2 was requested to create a thread by the attachment facility for the user. This does not include allocations for DB2 system agents.

Field Name: QTALLOCA

This is an exception field.

PLAN ALLOCATION SUCCESSFUL

The number of successful plan allocation attempts.

The cause of plan allocation failure could be plan unavailability or attempting to allocate a nonexistent plan.

Field Name: QTALLOC

PACKAGE ALLOCATION ATTEMPT

The number of attempts to allocate a package.

Field Name: QTPKALLA

PACKAGE ALLOCATION SUCCESS

The number of successful package allocation attempts.

Background and Tuning Information

Package allocation failure can occur when a package is unavailable or does not exist.

A high count of the number of packages unsuccessfully allocated (QTPKALLA - QTPKALL) typically occurs when a package list with multiple collections is used and frequently-used packages are found in the back end rather than in the front end of a package list. For example, when a package is found in the tenth collection, QTPKALLA is incremented by 10, one for each collection searched, but QTPKALL is incremented by 1.

A high number of packages unsuccessfully allocated can be accompanied by a high count of the number of unsuccessful checks for package execute authority made using the package authorization check because an application entry was not found in the cache (QTPACNOT). In this case, placing frequently used packages in the front end of a package list would reduce the number of Buffer Manager Getpages to the catalog/directory tablespaces.

Field Name: QTPKALL

PLANS BOUND

The number of plans successfully bound and kept for future agent allocations.

This field represents the sum of successful BIND ADD (QTBINDA) and successful BIND REPLACE (QTBINDR) commands. This counter is not incremented for BIND subcommands with no plan ID specified, as identified by QTTESTB. Note that QTBINDA + QTBINDR is not necessarily equal to this field. It is equal only if all BIND ADD and BIND REPLACE subcommands issued are successful.
Plan/Packet Activity

Field Name: QTPLNBD

BIND ADD SUBCOMMANDS
The number of successful and unsuccessful BIND ADD subcommands issued.
The sum of QTBINDA, QTBINDR, and QTTESTB equals the total number of BIND subcommands.
Field Name: QTBINDA

BIND REPLACE SUBCOMMANDS
The number of successful and unsuccessful BIND REPLACE subcommands issued.
Field Name: QTBINDR

TEST BINDS NO PLAN-ID
The number of BIND subcommands issued without a plan ID.
Field Name: QTTESTB

PACKAGES BOUND
The number of packages bound and kept for future package allocations.
It is the sum of successful BIND ADD PACKAGE and BIND REPLACE PACKAGE subcommands, but only if all these commands are really issued successfully.
Field Name: QTPKGBD

BIND ADD PACKAGE SUBCOMMAND
The number of successful and unsuccessful BIND ADD PACKAGE subcommands issued.
Field Name: QTBINDPA

BIND REPLACE PACKAGE SUBCOMMAND
The number of successful and unsuccessful BIND REPLACE PACKAGE subcommands issued.
Field Name: QTBINDPR

AUTOMATIC BIND ATTEMPTS
The number of attempts to autobind a plan. This occurs when the plan was invalidated by modifications to the declarations of the data referenced by the programs bound as part of the plan. For example, dropping an index when it is used in the plan results in automatic bind.
Field Name: QTABINDA

AUTOMATIC BINDS SUCCESSFUL
The number of plans successfully autobound.
Field Name: QTABIND

AUTO.BIND INVALID RES. IDS
The number of requests to allocate a nonexistent plan or package. This is the number of all plan and package allocation attempts that failed because the resource was unavailable or the object did not exist.
Field Name: QTINVRID

AUTO.BIND PACKAGE ATTEMPTS

The number of attempts to autobind a package.

Background and Tuning Information

If YES was specified, or defaulted, for autobind on DB2 install panel DSNTIPB, an autobind occurs when a plan or package:

• Is invalid because declarations of the data referenced by the program or package were modified. For example, when an index used in a package is dropped, an automatic bind occurs when the package is run for the first time after the index was dropped.
• Was bound in a later release and is used in a previous release for the first time.
• Was used in a previous release but is later remigrated and used in a later release for the first time.

Field Name: QTAUTOBA

This is an exception field.

AUTO.BIND PACKAGES SUCCESS

The number of packages successfully autobound.

Field Name: QTPKABND

This is an exception field.

REBIND SUBCOMMANDS

The number of REBIND subcommands issued. More than one plan can be rebound with a single REBIND subcommand. If the value in this field is 1, the number of plans you are attempting to rebind is shown in the Rebind - plan attempts field.

Field Name: QTREBIND

ATTEMPTS TO REBIND A PLAN

The number of attempts to rebind a plan. This number can be larger than the value shown in the Rebind - plan subcommands field because you can specify more than one plan in a single REBIND subcommand.

Field Name: QTRBINDA

PLANS REBOUND

The number of rebind attempts that completed successfully. This field is equal to the Rebind - Plan attempts field if all specified plans rebound successfully.

Field Name: QTPLNRBD

REBIND PACKAGE SUBCOMMANDS

The number of REBIND PACKAGE subcommands issued. More than one package can be rebound with a single subcommand. If the value in this field is 1, Rebind - package attempts shows the number of packages you are attempting to rebind.

Field Name: QTRBINDP

ATTEMPTS TO REBIND PACKAGE
Plan/Package Activity

The number of attempts to rebind a package. This can be larger than the value shown in Rebind package subcommands because you can rebind more than one package with a single command.

Field Name: QTRBNDPA

PACKAGES REBOUND

The number of packages successfully rebound. If all specified packages were rebound successfully, this field is equal to Rebind package attempts.

Field Name: QTPKGRBD

FREE PLAN SUBCOMMANDS

The number of FREE subcommands issued.

More than one plan can be freed with a single FREE subcommand. If this field is 1, then the number of plans you are trying to free is shown in ATTEMPTS TO FREE A PLAN.

Field Name: QTFREE

ATTEMPTS TO FREE A PLAN

The number of attempts to free a plan.

This value can be larger than FREE PLAN SUBCOMMANDS because multiple plan IDs can be specified in a single FREE subcommand.

Field Name: QTFREEA

PLANS FREED

The number of times a plan was successfully freed.

Freeing a plan can fail if someone else is using the plan and holds a lock on it.

Field Name: QTPLNFRD

FREE PACKAGE SUBCOMMANDS

The number of FREE PACKAGE subcommands issued.

More than one package can be freed with a single FREE subcommand. If the value in this field is 1, then the number of packages you are attempting to free is shown in ATTEMPTS TO FREE A PACKAGE.

Field Name: QTFREEP

ATTEMPTS TO FREE A PACKAGE

The number of attempts to free a package. This number can be larger than FREE PACKAGE SUBCOMMANDS because you can free several packages with a single command.

Field Name: QTFREEAP

PACKAGES FREED

The number of times a package was successfully freed. If all the specified packages were freed successfully, the value of this field is equal to ATTEMPTS TO FREE A PACKAGE.

Field Name: QTPKGFRD
Query Parallelism

This topic shows detailed information about “Statistics - Query Parallelism”.

This block shows information about query parallelism used by DB2 to perform parallel operations in SQL query processing. Dependent on various settings, DB2 may create parallel groups for a query where each parallel group consists of a set of tasks or I/O operations that can be executed in parallel. The degree of parallelism is the number of parallel tasks or I/O operations that DB2 determines. It can be used for the operations on the parallel group.

The DB2 users can limit the maximum number of parallel operations to reduce the resource consumption of a parallelism environment, and even DB2 may decide during execution time to reduce the planned degree of parallelism in order to respond to system limitations.

Statistics - Query Parallelism

The field labels shown in the following sample layout of “Statistics - Query Parallelism” are described in the following section.

<table>
<thead>
<tr>
<th>QUERY PARALLELISM</th>
<th>QUANTITY /SECOND /THREAD /COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAX DEGREE - ESTIMATED</td>
<td>0.00 N/A N/A N/A</td>
</tr>
<tr>
<td>MAX DEGREE - PLANNED</td>
<td>0.00 N/A N/A N/A</td>
</tr>
<tr>
<td>MAX DEGREE - EXECUTED</td>
<td>0.00 N/A N/A N/A</td>
</tr>
<tr>
<td>PARALLEL GROUPS EXECUTED</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>RAN AS PLANNED</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>RAN REDUCED-STORE</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>RAN REDUCED-NEGOTIATION</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>SEQUENTIAL-CURSOR</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>SEQUENTIAL-NO ESA</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>SEQUENTIAL-NO BUFFER</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>SEQUENTIAL-ENCLAVE SER.</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>SEQUENTIAL-AUTONOMOUS PROC</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>SEQUENTIAL-NEGOTIATION</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>ONE DB2 - COORDINATOR = NO</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>ONE DB2 - ISOLATION LEVEL</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>ONE DB2 - DCL TTABLE</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>MEMBER SKIPPED (%)</td>
<td>N/C</td>
</tr>
<tr>
<td>REFORM PARAL-CONFIG CHANGED</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>REFORM PARAL-NO BUFFER</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
</tbody>
</table>

MAX DEGREE - ESTIMATED

The maximum degree of parallelism estimated for a parallel group at bind time based on the cost formula. If the parallel group contains a host variable or parameter marker, then bind time will estimate the parallel group degree based on a valid assumption value.

Field Name: QXMAXESTIDG

MAX DEGREE - PLANNED

The maximum degree of parallelism planned for a parallel group. It is the ideal parallel group degree obtained at execution time after the host variable or parameter marker value is "plug-in" and before buffer pool negotiation and system negotiation are performed.

Field Name: QXMAXPLANIDG

MAX DEGREE - EXECUTED

Chapter 49. Statistics Report and Trace Blocks
Query Parallelism

The maximum degree of parallelism executed among all parallel groups to indicate the extent to which queries were processed in parallel.

Field Name: QXMAXDEG

PARALLEL GROUPS EXECUTED

The total number of parallel groups executed.

Field Name: QXTOTGRP

RAN AS PLANNED

The total number of parallel groups that executed in the planned parallel degree. This field is incremented by one for each parallel group that executed in the planned degree of parallelism (as determined by DB2).

Field Name: QXNORGRP

RAN REDUCED-STORAGE

The total number of parallel groups that did not reach the planned parallel degree because of a lack of storage space or contention on the buffer pool.

The exception field name is QXREDGRP.

Background and Tuning Information

If this field is not 0, increase the size of the current buffer pool using the ALTER BUFFERPOOL command or use the ALTER TABLESPACE command to assign table spaces accessed by this query to a different buffer pool.

Field Name: QXREDGRP

This is an exception field.

RAN REDUCED-NEGOTIATION

The total number of parallel groups that did not reach the planned parallel degree due to system negotiation result of system stress level.

Field Name: QXSTOREDGRP

SEQUENTIAL-CURSOR

The total number of parallel groups that fell back to sequential mode due to a cursor that can be used by UPDATE or DELETE.

Field Name: QXDEGCUR

SEQUENTIAL-NO ESA

The total number of parallel groups that fell back to sequential mode due to a lack of ESA sort support.

Field Name: QXDEGESA

SEQUENTIAL-NO BUFFER

The total number of parallel groups that fell back to sequential mode due to a storage shortage or contention on the buffer pool.

The exception field name is QXDEGBUF.

Field Name: QXDEGBUF

SEQUENTIAL-ENCLAVE SER.
Query Parallelism

The total number of parallel groups that executed in sequential mode due to the unavailability of MVS/ESA enclave services.

**Field Name:** QXDEGENC

This is an *exception* field.

**SEQUENTIAL-AUTONOMOUS PROC**

The total number of parallel groups that fell back to sequential mode under an autonomous procedure.

**Field Name:** QXDEGAT

**SEQUENTIAL-NEGOTIATION**

The total number of parallel groups that fell back to sequential mode due to system negotiation result of system stress level.

**Field Name:** QXSTODGNGRP

**ONE DB2 - COORDINATOR = NO**

The total number of parallel groups executed on a single DB2 subsystem due to the COORDINATOR subsystem value being set to NO. When the statement was bound, the COORDINATOR subsystem value was set to YES. This situation can also occur when a package or plan is bound on a DB2 subsystem with COORDINATOR=YES, but is run on a DB2 subsystem with COORDINATOR=NO.

**Field Name:** QXCOORNO

**ONE DB2 - ISOLATION LEVEL**

The total number of parallel groups executed on a single DB2 subsystem due to repeatable-read or read-stability isolation.

**Field Name:** QXISORR

**ONE DB2 - DCL TTABLE**

The number of parallel groups in a query block that were downgraded to CPU parallelism because they referenced a UDF and a declared temporary table was detected at execution time.

DB2 enforces execution on a single DB2 (CPU parallelism), in this instance, because it cannot determine at incremental bind time for the statement whether the UDF will reference the declared temporary table. Other parallel groups in the same statement are not necessarily downgraded.

**Field Name:** QXDEGDTT

**MEMBER SKIPPED (%)**

The percentage of parallel groups that were not distributed over the data sharing group because one or more DB2 members did not have enough buffer pool storage. This only applies to parallel groups that were intended to run in sysplex query parallelism.

**Background and Tuning Information**

This percentage is only recorded when the buffer pool is defined to allow for parallelism. For example, if VPXPSEQT=0 on an assistant, DB2 does not send parallel work there, and the percentage is not increased.

**Field Name:** SXXCRAT

**REFORM PARAL-CONFIG CHANGED**
Query Parallelism

The total number of parallel groups where DB2 reformulated the parallel portion of the access path because of a change in the number of active members, or because of a change of processor models on which they run, from bind time to run time. This counter is incremented only on the parallelism coordinator at run time.

Field Name: QXREPOP1

REFORM PARAL-NO BUFFER

The total number of parallel groups in which DB2 reformulated the parallel portion of the access path because there were insufficient buffer-pool resources. This counter is incremented only at the parallelism coordinator at run time.

Field Name: QXREPOP2
Real and Auxiliary Storage for DBM1

This topic shows detailed information about “Statistics - Real and Auxiliary Storage for DBM1”.

Statistics - Real and Auxiliary Storage for DBM1

The field labels shown in the following sample layout of “Statistics - Real and Auxiliary Storage for DBM1” are described in the following section.

<table>
<thead>
<tr>
<th>REAL AND AUXILIARY STORAGE FOR DBM1</th>
<th>QUANTITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>REAL STORAGE IN USE (MB)</td>
<td>8403.51</td>
</tr>
<tr>
<td>31 BIT IN USE (MB)</td>
<td>160.28</td>
</tr>
<tr>
<td>64 BIT IN USE (MB)</td>
<td>8243.23</td>
</tr>
<tr>
<td>64 BIT THREAD AND SYSTEM ONLY (MB)</td>
<td>394.61</td>
</tr>
<tr>
<td>HWM 64 BIT REAL STORAGE IN USE (MB)</td>
<td>8243.23</td>
</tr>
<tr>
<td>AVERAGE THREAD FOOTPRINT (MB)</td>
<td>0.57</td>
</tr>
<tr>
<td>AUXILIARY STORAGE IN USE (MB)</td>
<td>0.00</td>
</tr>
<tr>
<td>31 BIT IN USE (MB)</td>
<td>0.00</td>
</tr>
<tr>
<td>64 BIT IN USE (MB)</td>
<td>0.00</td>
</tr>
<tr>
<td>64 BIT THREAD AND SYSTEM ONLY (MB)</td>
<td>0.00</td>
</tr>
<tr>
<td>HWM 64 BIT AUX STORAGE IN USE (MB)</td>
<td>0.00</td>
</tr>
</tbody>
</table>

REAL STORAGE IN USE (MB)

The amount of real storage in use for 31-bit and 64-bit private pools.

Field Name: SW0225RL

31 BIT IN USE (MB)

The amount of real storage in use for 31-bit private pools.

Note: This value is available from z/OS V1.11.

Field Name: S225RL31

64 BIT IN USE (MB)

The amount of real storage in use for 64-bit private pools.

Note: This value is available from z/OS V1.11.

Field Name: S225VPR

64 BIT THREAD AND SYSTEM ONLY (MB)

The amount of real storage in use for 64-bit private pools that does not include buffer pool storage.

Note: This field is available in z/OS 1.10 (and maintenance) or later.

Field Name: S225PSR

HWM 64 BIT REAL STORAGE IN USE (MB)

High water mark of real storage in use for 64-bit private pools.

Note: This value is available from z/OS V1.11.

Field Name: S225GPR

AVERAGE THREAD FOOTPRINT (MB)
Real and Auxiliary Storage for DBM1

The current average real storage in use for private DBM1 storage of active user threads (allied threads + active and pooled DBATs).

Field Name: S225DTFR

AUXILIARY STORAGE IN USE (MB)

The amount of auxiliary storage in use for 31-bit and 64-bit private pools.

Field Name: SW0225AX

31 BIT IN USE (MB)

The amount of auxiliary storage in use for 31-bit private pools.

Note: This value is available from z/OS V1.11.

Field Name: S225AX31

64 BIT IN USE (MB)

The amount of auxiliary storage in use for 64-bit private pools.

Note: This value is available from z/OS V1.11.

Field Name: S225VAS

64 BIT THREAD AND SYSTEM ONLY (MB)

The amount of auxiliary storage in use for 64-bit private pools that does not include buffer pool storage.

Note: This field is available in z/OS 1.10 (and maintenance) or later.

Field Name: S225PSA

HWM 64 BIT AUX STORAGE IN USE (MB)

High water mark of auxiliary storage in use for 64-bit private pools.

Note: This value is available from z/OS V1.11.

Field Name: S225GAS
Real and Auxiliary Storage for DIST (DB2 10 or later)

This topic shows detailed information about “Statistics - Real and Auxiliary Storage for DIST (DB2 10 or later)”.

Statistics - Real and Auxiliary Storage for DIST (DB2 10 or later)

The field labels shown in the following sample layout of “Statistics - Real and Auxiliary Storage for DIST (DB2 10 or later)” are described in the following section.

<table>
<thead>
<tr>
<th>REAL AND AUXILIARY STORAGE FOR DIST</th>
<th>QUANTITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>REAL STORAGE IN USE (MB)</td>
<td>86.89</td>
</tr>
<tr>
<td>31 BIT IN USE (MB)</td>
<td>59.47</td>
</tr>
<tr>
<td>64 BIT IN USE (MB)</td>
<td>27.41</td>
</tr>
<tr>
<td>64 BIT THREAD AND SYSTEM ONLY (MB)</td>
<td>27.40</td>
</tr>
<tr>
<td>HWM 64 BIT REAL STORAGE IN USE (MB)</td>
<td>27.41</td>
</tr>
<tr>
<td>AVERAGE DBAT FOOTPRINT (MB)</td>
<td>0.09</td>
</tr>
<tr>
<td>AUXILIARY STORAGE IN USE (MB)</td>
<td>0.00</td>
</tr>
<tr>
<td>31 BIT IN USE (MB)</td>
<td>0.00</td>
</tr>
<tr>
<td>64 BIT IN USE (MB)</td>
<td>0.00</td>
</tr>
<tr>
<td>64 BIT THREAD AND SYSTEM ONLY (MB)</td>
<td>0.00</td>
</tr>
<tr>
<td>HWM 64 BIT AUX STORAGE IN USE (MB)</td>
<td>0.00</td>
</tr>
</tbody>
</table>

REAL STORAGE IN USE (MB)

The amount of real storage in use for 31-bit and 64-bit private pools.

Field Name: SW0225RL

31 BIT IN USE (MB)

The amount of real storage in use for 31-bit private pools.

Note: This value is available from z/OS V1.11.

Field Name: S225RL31

64 BIT IN USE (MB)

The amount of real storage in use for 64-bit private pools.

Note: This value is available from z/OS V1.11.

Field Name: S225VPR

64 BIT THREAD AND SYSTEM ONLY

The amount of real storage in use for 64-bit private pools that does not include buffer pool storage.

Note: This field is available in z/OS 1.10 (and maintenance) or later.

Field Name: S225PSR

HWM 64 BIT REAL STORAGE IN USE (MB)

High water mark of real storage in use for 64-bit private pools.

Note: This value is available from z/OS V1.11.

Field Name: S225GPR

AVERAGE DBAT FOOTPRINT
Real and Auxiliary Storage for DIST

The current average real storage in use for private DIST storage of active and pooled DBATs.

Field Name: S225DDFR

AUXILIARY STORAGE IN USE (MB)

The amount of auxiliary storage in use for 31-bit and 64-bit private pools.

Field Name: SW0225AX

31 BIT IN USE (MB)

The amount of auxiliary storage in use for 31-bit private pools.

Note: This value is available from z/OS V1.11.

Field Name: S225AX31

64 BIT IN USE (MB)

The amount of auxiliary storage in use for 64-bit private pools.

Note: This value is available from z/OS V1.11.

Field Name: S225VAS

64 BIT THREAD AND SYSTEM ONLY

The amount of auxiliary storage in use for 64-bit private pools that does not include buffer pool storage.

Note: This field is available in z/OS 1.10 (and maintenance) or later.

Field Name: S225PSA

HWM 64 BIT AUX STORAGE IN USE (MB)

High water mark of auxiliary storage in use for 64-bit private pools.

Note: This value is available from z/OS V1.11.

Field Name: S225GAS
Real Storage in Use - Summary

This topic shows detailed information about “Statistics - Real Storage in Use - Summary”.

Statistics - Real Storage in Use - Summary

The field labels shown in the following sample layout of “Statistics - Real Storage in Use - Summary” are described in the following section.

<table>
<thead>
<tr>
<th>REAL STORAGE IN USE - SUMMARY</th>
<th>QUANTITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>31/64-BIT PRIVATE (DBM1) (MB)</td>
<td>145.25</td>
</tr>
<tr>
<td>31/64-BIT PRIVATE (DIST) (MB)</td>
<td>5.34</td>
</tr>
<tr>
<td>64-BIT SHARED THREAD AND SYSTEM (MB)</td>
<td>17.25</td>
</tr>
<tr>
<td>64-BIT SHARED STACK (MB)</td>
<td>13.97</td>
</tr>
<tr>
<td>64-BIT COMMON (MB)</td>
<td>3.29</td>
</tr>
<tr>
<td>TOTAL REAL STORAGE IN USE (MB)</td>
<td>185.11</td>
</tr>
</tbody>
</table>

31/64-BIT PRIVATE (DBM1) (MB)

The amount of real storage in use for 31-bit and 64-bit private pools.

Field Name: SW0225RL

31/64-BIT PRIVATE (DIST) (MB)

The amount of real storage in use for 31-bit and 64-bit private pools.

Field Name: SW0225RL

64-BIT SHARED THREAD AND SYSTEM (MB)

The amount of real storage in use for 64-bit shared storage. This does not include shared stack storage. This is recorded at the subsystem level.

Note: This field is available in z/OS 1.10 (and maintenance) or later.

Field Name: S225SSR

64-BIT SHARED STACK (MB)

The amount of real storage in use for 64-bit shared stack storage. This is recorded at the subsystem level.

Note: This field is available in z/OS 1.10 (and maintenance) or later.

Field Name: S225KSR

64-BIT COMMON (MB)

The amount of real storage in use for 64-bit common storage. This is recorded at the subsystem level.

Note: This field is available in z/OS 1.10 (and maintenance) or later.

Field Name: S225CSR

TOTAL REAL STORAGE IN USE (MB)

The total amount of real storage in use.

Field Name: S225RLTL
RID List Processing

This topic shows detailed information about “Statistics - RID List Processing”.

The RID pool is used for:
- List prefetch
- Multiple index access
- Hybrid joins

DB2 uses a matching index scan to collect those record identifiers (RID) that match the selection criteria and places them in a list in the RID pool. The list is sorted by page number, which is contained in the RID. DB2 then uses the sorted list to access the table by reading up to 32 pages per I/O and attempting to read ahead one block of 32 pages before use.

The RID pool is allocated dynamically as it is needed. The maximum size of the pool is determined by the ZPARM MAXRBLK.

Statistics - RID List Processing

The field labels shown in the following sample layout of “Statistics - RID List Processing” are described in the following section.

<table>
<thead>
<tr>
<th>RID LIST PROCESSING</th>
<th>QUANTITY /SECOND /THREAD /COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUCCESSFUL</td>
<td>8680.8K 4667.07 N/C 7.28</td>
</tr>
<tr>
<td>NOT USED-NO STORAGE</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
<tr>
<td>NOT USED-MAX LIMIT</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
<tr>
<td>MAX RID BLOCKS ALLOCATED</td>
<td>147.00 N/A N/A N/A</td>
</tr>
<tr>
<td>CURRENT RID BLOCKS ALLOCAT.</td>
<td>7.74 N/A N/A N/A</td>
</tr>
<tr>
<td>MAX RID BLOCKS OVERFLOWED</td>
<td>0.00 N/A N/A N/A</td>
</tr>
<tr>
<td>CURRENT RID BLOCKS OVERFL.</td>
<td>0.00 N/A N/A N/A</td>
</tr>
<tr>
<td>TERMINATED-NO STORAGE</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
<tr>
<td>TERMINATED-EXCEED RDS LIMIT</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
<tr>
<td>TERMINATED-EXCEED DM LIMIT</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
<tr>
<td>TERMINATED-EXCEED PROC.LIM.</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
<tr>
<td>OVERFLOWED-NO STORAGE</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
<tr>
<td>OVERFLOWED-MAX LIMIT</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
<tr>
<td>INTERRUPTED-NO STORAGE</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
<tr>
<td>INTERRUPTED-MAX LIMIT</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
<tr>
<td>SKIPPED-INDEX KNOWN</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
</tbody>
</table>

SUCCESSFUL

The number of times RID list (also called RID pool) processing is used.

During RID (RECORD ID) list processing, DB2 uses an index to produce a list of candidate RIDs, which is called a RID list. The RID list can be sorted and intersected (ANDed) or unioned (ORed) with other RID lists before actually accessing the data pages. RID list processing is used for a single index (index access with list prefetch) or for multiple indexes (multiple index access), which is when the RID lists are ANDed and ORed.

This field is incremented once for a given table access when RID list processing is used for index access with list prefetch, for multiple index access, or for both. For multiple index access, if a final RID list is obtained through ANDing and ORing of RID lists, the counter is incremented once, even if not all indexes were used by the RIDs in the multiple index access.
RID List Processing

Background and Tuning Information
A nonzero value in this field indicates that DB2 has used list prefetch. If this is the case, check the access path selection.

Field Name: QXMIAP
This is an exception field.

NOT USED-NO STORAGE
The number of times DB2 detected that no storage was available to hold a list of RIDs during a given RID list process involving one index (single index access with list prefetch) or multiple indexes (multiple index access).

This field can be incremented during retrieval, sorting, ANDing, and ORing of RID lists for index access with list prefetch (single index). For single index access, this field can only be incremented once per access. For multiple index access, it can be incremented for every index involved in the ANDing and ORing of RID lists.

Field Name: QXNSMIAP
This is an exception field.

NOT USED-MAX LIMIT
The number of times DB2 detected that a RID list exceeded one or more internal limits during a given RID list (or RID pool) process involving one index (single index access with list prefetch) or multiple indexes (multiple index access). The internal limits include the physical limitation of the number of RIDs a RID list can hold and threshold values for the retrieval, ORing, and ANDing of RIDs.

For index access with list prefetch (single index), this field can only be incremented during RID list retrieval. For multiple index access, this field can be incremented during RID list retrieval, ANDing, and ORing. This counter reflects the number of times internal limits or threshold values were exceeded for the RID lists obtained directly from an index as well as for RID lists derived during the ANDing and ORing process.

Background and Tuning Information
Before you increase the RID list storage size, investigate the cause of the failure using the statistics record or the performance trace. You can specify the desired size for the RID list (within the range of 16 KB to 1000 MB) on the DB2 installation panel DSNTIPC.

Field Name: QXMRMIAP
This is an exception field.

MAX RID BLOCKS ALLOCATED
The highest number of RID blocks in use at any time since DB2 startup. This is a high-water mark.

Field Name: QISTRHIG
This is an exception field.

CURRENT RID BLOCKS ALLOCATED
The number of RID blocks currently in use (snapshot value).

Field Name: QISTRCUR
RID List Processing

This is an exception field.

MAX RID BLOCKS OVERFLOWED

The maximum number of RID blocks overflowed (stored) to a work file at any time since DB2 start. This is a high-water mark.

Field Name: QISTWFRHIG

CURRENT RID BLOCKS OVERFL.

The number of RID blocks currently residing in work file storage. This is a snapshot value.

Field Name: QISTWFRCUR

TERMINATED-NO STORAGE

The number of times RID list processing was not used because DBM1 storage was exhausted.

Background and Tuning Information

This failure occurs when the DBM1 storage limit is reached.

Field Name: QISTRSTG

This is an exception field.

TERMINATED-EXCEED RDS LIMIT

The number of times RID list processing terminated because the number of RIDs that can fit into the guaranteed number of RID blocks was greater than the maximum limit (25% of table size).

Background and Tuning Information

Ideally, this value should be 0.

The matching index scan part of the RID list processing scanned more than 25% of the index. RID list processing is then terminated, the index scan is abandoned and normally replaced by a tablespace scan.

Reasons for this are:

- Inaccurate or incomplete RUNSTATS statistics. To avoid this, you should collect all statistics on a regular basis, especially simple and correlated column statistics. Using RUNSTATS with SHRLEVEL(CHANGE) does not prevent access to data.
- Optimizer error. In this instance, you could disable RID list processing by adding the clause OPTIMIZE FOR 1 ROW to the SQL statement, or force the access path to index only by adding the necessary columns to the index.

Field Name: QISTRLLM

This is an exception field.

TERMINATED-EXCEED DM LIMIT

The number of times a RID list processing operation terminated because the number of RID entries was greater than the physical limit of approximately 26 million RIDs.

Field Name: QISTRPLM

This is an exception field.

TERMINATED-EXCEED PROC.LIM.
The number of times the maximum RID pool storage was exceeded.

The size is determined by the installation parameter RID POOL SIZE (DB2 install panel DSNTIPC). It can be 0, or between 128 KB and 10 GB. The general formula for calculating the RID pool size is:

(Number of concurrent RID processing activities) x (average number of RIDs) x 2 x (5 bytes per RID).

Field Name: QISTRMAX

This is an exception field.

OVERFLOWED-NO STORAGE

The number of times a RID list was overflowed to a work file because no RID pool storage was available to hold the list of RIDs.

Field Name: QXWFRIDS

OVERFLOWED-MAX LIMIT

The number of times a RID list was overflowed to a work file because the number of RIDs exceeded one or more internal limits.

Field Name: QXWFRIDT

INTERRUPTED-NO STORAGE

The number of times a RID list append for a hybrid join was interrupted because no RID pool storage was available to hold the list of RIDs.

Field Name: QXHJINCS

INTERRUPTED-MAX LIMIT

The number of times a RID list append for a hybrid join was interrupted because the number of RIDs exceeded one or more internal limits.

Field Name: QXHJINCT

SKIPPED-INDEX KNOWN

The number of times a RID list retrieval for multiple index access was skipped because it was not necessary due to DB2 being able to predetermine the outcome of index ANDing or ORing.

Field Name: QXRSMIAP
This topic shows detailed information about “Statistics - ROWID”.

**Statistics - ROWID**

The field labels shown in the following sample layout of “Statistics - ROWID” are described in the following section.

<table>
<thead>
<tr>
<th>ROW ID</th>
<th>QUANTITY</th>
<th>/SECOND</th>
<th>/THREAD</th>
<th>/COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>DIRECT ACCESS</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>INDEX USED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>TABLE SPACE SCAN USED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
</tbody>
</table>

**DIRECT ACCESS**

The number of times that direct row access was successful.

*Field Name:* QXROIMAT

**INDEX USED**

The number of times that direct row access failed and an index was used to find a record.

**Background and Tuning Information**

This can happen, for example, when a REORG is performed between the read of the ROWID column and the use of the host variable in the WHERE clause of the SQL statement. This causes the RID value in the host variable to be incorrect.

*Field Name:* QXROIIDX

**TABLE SPACE SCAN USED**

The number of times that an attempt to use direct row access reverted to using a table-space scan because DB2 was unable to use a matching index scan.

**Background and Tuning Information**

Ideally, this value should be 0.

Table-space scans can happen, for example, when a REORG is performed between the read of the ROWID column and the use of the host variable in the WHERE clause of the SQL statement. This causes the RID value in the host variable to be incorrect. DB2 first tries a matching-index scan before using a table-space scan.

To avoid table space scans, you can force the access path of an unsuccessful direct row access to use a matching index scan on the primary-index key by adding PKCOL to the WHERE clause in the SQL statement. 

```
WHERE ROWIDCOL=:HVROWID AND PKCOL=:HVPK
```

*Field Name:* QXROITS
Short-on-Storage Metrics

This topic shows detailed information about “Statistics - Short-on-Storage Metrics”.

Statistics - Short-on-Storage Metrics

The field labels shown in the following sample layout of “Statistics - Short-on-Storage Metrics” are described in the following section.

<table>
<thead>
<tr>
<th>SHORT-ON-Storage Metrics</th>
<th>QUANTITY /SECOND /THREAD /COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>FULL SYSTEM CONTRACTIONS</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>CRITICAL SHORTAGES</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
<tr>
<td>ABENDS DUE TO SHORTAGES</td>
<td>0.00 0.00 0.00 0.00</td>
</tr>
</tbody>
</table>

FULL SYSTEM CONTRACTIONS

The number of full system contractions.

Field Name: QSSTCONT

CRITICAL SHORTAGES

The number of critical storage shortages after contraction.

Field Name: QSSTCRIT

ABENDS DUE TO SHORTAGES

The number of abends due to local storage shortage.

Field Name: QSSTABND
This topic shows detailed information about “Statistics - Stored Procedures”.

**Statistics - Stored Procedures**

The field labels shown in the following sample layout of “Statistics - Stored Procedures” are described in the following section.

<table>
<thead>
<tr>
<th>STORED PROCEDURES</th>
<th>QUANTITY /SECOND /THREAD /COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>CALL STATEMENT EXECUTED</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
<tr>
<td>PROCEDURE ABENDED</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
<tr>
<td>CALL STATEMENT TIMED OUT</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
<tr>
<td>CALL STATEMENT REJECTED</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
</tbody>
</table>

**CALL STATEMENTS EXECUTED**

The number of SQL CALL statements executed.

*Field Name:* QXCALL

**PROCEDURE ABENDED**

The number of times a stored procedure terminated abnormally.

*Field Name:* QXCALLAB

**CALL STATEMENT TIMED OUT**

The number of times an SQL call timed out waiting to be scheduled.

*Field Name:* QXCALLTO

**CALL STATEMENT REJECTED**

The number of times an SQL CALL statement was rejected due to the procedure being in the STOP ACTION(REJECT) state.

*Field Name:* QXCALLRJ
Subsytem Services

This topic shows detailed information about “Statistics - Subsystem Services”.

**Statistics - Subsystem Services**

The field labels shown in the following sample layout of “Statistics - Subsystem Services” are described in the following section.

<table>
<thead>
<tr>
<th>SUBSYSTEM SERVICES</th>
<th>QUANTITY /SECOND</th>
<th>THREAD</th>
<th>COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>IDENTIFY</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
</tr>
<tr>
<td>CREATE THREAD</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
</tr>
<tr>
<td>SIGNON</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
</tr>
<tr>
<td>TERMINATE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
</tr>
<tr>
<td>ROLLBACK</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
</tr>
<tr>
<td>COMMIT PHASE 1</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
</tr>
<tr>
<td>COMMIT PHASE 2</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
</tr>
<tr>
<td>READ ONLY COMMIT</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
</tr>
<tr>
<td>UNITS OF RECOVERY INDOUBT</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
</tr>
<tr>
<td>UNITS OF REC.INDBT RESOLVED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
</tr>
<tr>
<td>SYNCHS(SINGLE PHASE COMMIT)</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
</tr>
<tr>
<td>QUEUED AT CREATE THREAD</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
</tr>
<tr>
<td>SUBSYSTEM ALLIED MEMORY EOT</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
</tr>
<tr>
<td>SUBSYSTEM ALLIED MEMORY EOM</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
</tr>
<tr>
<td>SYSTEM EVENT CHECKPOINT</td>
<td>17.00</td>
<td>0.09</td>
<td>N/C</td>
</tr>
<tr>
<td>HIGH WATER MARK IDBACK</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
</tr>
<tr>
<td>HIGH WATER MARK IDFORE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
</tr>
<tr>
<td>HIGH WATER MARK CTHREAD</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
</tr>
</tbody>
</table>

**IDENTIFY**

The number of successful connections to DB2 from an allied address space (TSO, BATCH, CICS, IMS, CAF, or UTILITY).

**Field Name:** Q3STIDEN

**CREATE THREAD**

The number of successful create thread requests. It does not include DBATs.

A thread is required before an application can use SQL. When established, a thread can have one or more secondary authorization IDs.

A thread is needed to perform any DB2 activity. For example, a thread is needed to run a DB2 utility to perform an IFI request such as READS, or to process a DB2 command such as -DISPLAY THREAD. However, a thread is not created if the command failed because of a syntax error.

**Background and Tuning Information**

Thread reuse can help improve performance.

The term *thread reuse* only applies to IMS and CICS attachments. In the case of the TSO attachment facility and the call attachment facility (CAF), threads cannot be reused, because the threads are allocated to the user address space.

Thread reuse should be considered in the following cases:

- If transaction volume is high:

  High volume transactions should achieve a high percentage of thread reuse. If threads are reused on low volume transactions, the number of
threads needed increases because these threads are not automatically terminated by IMS when not being used. This may result in too many idle threads for the level of the DB2 workload. Under CICS, protected threads are terminated after about 45 seconds if no transaction eligible to reuse the thread has been received.

- If thread creation cost is significant:
  As a rule of thumb, more than 5% of the total CPU cost of transaction processing is considered significant.

The ACQUIRE and RELEASE parameters of BIND should be specified to minimize the thread creation cost, while providing the needed concurrency:

- If most of the application plan's SQL statements are executed, then ACQUIRE(ALLOCATE) is cheaper than ACQUIRE(USE).
- If only a small number of the SQL statements are executed, ACQUIRE(USE) becomes cheaper and improves concurrency, because the required resources are only acquired (locked) when the plan actually references (uses) them. An example would be a generalized plan used by many different transactions. It would contain multiple logic paths referencing different tables.
  Note that, when packages are involved, ACQUIRE(USE) is always implicitly used.
- Concurrency in thread reuse is based on page locking provided by the IS and IX intent locks, whose duration is governed by ACQUIRE and RELEASE of BIND.
  RELEASE(DEALLOCATE) is strongly recommended for thread-reuse transactions to reduce transaction CPU time.

When thread reuse is implemented, monitor the EDM pool. It should be sufficient in size to accommodate expanding plans where the next transaction requires additional plan sections over those that are already part of the plan.

**Field Name:** Q3STCTHD

This is an *exception* field.

**SIGNON**

The number of signons that identified a new user of an existing thread for IMS and CICS.

This field is valid only for CICS and IMS (not valid for TSO, CAF, or UTILITY).

The initial signon does not perform an authorization check because the thread does not exist yet, but a resignon can.

**Background and Tuning Information**

If the number of signons is greater than the number of create thread occurrences, some threads have been reused. In the case of the TSO attachment facility and the call attachment facility (CAF), there is no sign-on, because the user is identified when the TSO address space is connected.

**Field Name:** Q3STSIGN

This is an *exception* field.
Subsystem Services

The number of time threads that terminated successfully.

This number does not agree with the create thread count because each level of a thread’s access (IDENTIFY, SIGNON, and CREATE THREAD) must be terminated.

Background and Tuning Information

The value of this field is usually greater than the number of create thread occurrences, because it also includes the termination of connections to DB2 (IDENTIFY) and other internal counts.

Field Name: Q3STTERM

ROLLBACK

The number of times a unit of recovery was successfully rolled back. Some reasons for a rollback include:

- Application program abend
- Application rollback request
- Application deadlock on database records
- Application canceled by operator
- Thread abend due to resource shortage

This number also includes successfully aborted agents associated with threads that use the Recoverable Resource Manager Services Attach Facility (RRSAF).

Field Name: Q3STABRT

This is an exception field.

COMMIT PHASE 1

The number of successful requests for commit phase 1 in a two-phase commit environment such as CICS or IMS. It includes successfully prepared agents associated with threads that use the Recoverable Resource Manager Services Attach Facility (RRSAF). It does not include successful single-phase commits or distributed two-phase commits.

Background and Tuning Information

IMS and CICS applications use the PREPARE and COMMIT sequence to commit work.

Field Name: Q3STPREP

This is an exception field.

COMMIT PHASE 2

The number of successful commit phase 2 in a two-phase environment such as CICS or IMS. It includes successfully committed agents associated with threads that use the Recoverable Resource Manager Services Attach Facility (RRSAF). It does not include successful single-phase commits or distributed two-phase commits.

Background and Tuning Information

IMS and CICS applications use the PREPARE and COMMIT sequence to commit work. A nonzero value for this field indicates that updates have occurred.

Field Name: Q3STCOMM
**Subsystem Services**

**READ ONLY COMMIT**

The number of read-only commits.

There are occasions when CICS or IMS invokes DB2 when no DB2 resource was altered since the completion of the last commit process. When this occurs, DB2 performs both phases of the two-phase commit during the first commit phase and records that the user or job is read-only in relation to its DB2 processing.

**Field Name:** Q3STRDON

**UNITS OF RECOVERY INDOUBT**

The number of indoubt units of recovery.

A unit of recovery is indoubt when a failure occurs after a successful prepare but before a successful commit. The failure can occur in the address space of the application, the transaction manager, DB2, or all of these. IMS and CICS applications use the prepare and commit sequence to commit work. Ideally, this value should be 0.

**Field Name:** Q3STINDT

**UNITS OF REC. INDBT RESOLVED**

The number of indoubt units of recovery successfully resolved, either automatically or manually. It includes successful indoubt resolutions for agents associated with threads that use the Recoverable Resource Manager Services Attach Facility (RRSAF).

A unit of recovery is indoubt when a failure occurs after a successful prepare but before a successful commit. This number should equal the number of units of recovery gone indoubt. If it is less, then some indoubt units of recovery might still exist.

**Field Name:** Q3STRIUR

**SYNCHS(SINGLE PHASE COMMIT)**

The total number of commits in a single-phase commit environment such as TSO, CAF, or UTILITY. IMS applications use the prepare-and-commit sequence; CICS applications use both the synchronized commit request and the prepare-and-commit sequence to commit work.

Note that DBATs executed on this location are not included. For DBAT statistics, see SINGLE PHASE COMMITs received on the DDF activity block.

**Field Name:** Q3STSYNC

**QUEUED AT CREATE THREAD**

The number of create thread requests queued (not including DBATs).

The total number of threads accessing data that can be allocated concurrently is the MAX USERS value on the installation panel DSNTIPE. Requests are queued when the MAX USERS value is exceeded. If no threads are queued during peak hours, the maximum number of threads might be set too high.

**Background and Tuning Information**

As a rule of thumb about 1% thread queuing is acceptable. When this is appreciably higher, increase the value of MAX USERS on the DB2 install panel DSNTIPE.
Subsystem Services

The combined maximum allowed for MAX USERS and MAX REMOTE ACTIVE cannot exceed 2000.

Field Name: Q3STCTHW
This is an exception field.

SUBSYSTEM ALLIED MEMORY EOT
The number of times non-DB2 tasks abended while connected to DB2.

Field Name: Q3STMEOT
This is an exception field.

SUBSYSTEM ALLIED MEMORY EOM
The number of times MVS deleted non-DB2 address space while connected to DB2.

Field Name: Q3STMEOM
This is an exception field.

SYSTEM EVENT CHECKPOINT
The number of checkpoints DB2 has taken.

A checkpoint is a point at which DB2 records internal status information to the DB2 log. This information is used in the recovery process if DB2 abends.

Background and Tuning Information

For Statistics reports only: A checkpoint is taken when the specified number of log records have been written. A checkpoint is also taken each time DB2 switches to a new active log data set. If the Statistics reporting period is 30 minutes and the value of this field is 15, then DB2 is taking checkpoints every 2 minutes.

If the data sets are too small or the value for LOGLOAD is too low, checkpoints occur too frequently. As a result, database writes do not perform efficiently. The frequency of DB2 checkpoints can be decreased by increasing the value of the DSNZPARM LOGLOAD (CHECKPOINT FREQ on the Tracing install panel).

Rule of thumb: In a production environment, DB2 should take checkpoints every 10 minutes or so.

The default value for LOGLOAD is 50000. The actual value that you choose is dependent on the volume and nature of the work performed by your DB2 subsystem. It is a trade-off between the performance efficiency of larger numbers and the longer time to restart DB2 when there is an abnormal termination.

Field Name: QWSDCCKPT
This is an exception field.

HIGH WATER MARK IDBACK
The maximum number of connections to a single instance from batch or TSO background tasks.

This is a high-water mark.

Field Name: Q3STHWIB
Subsystem Services

HIGH WATER MARK IDFORE

The maximum number of connections to a single instance from TSO foreground tasks.

This is a high water-mark.

Field Name: Q3STHWIF

HIGH WATER MARK CTHREAD

The highest number of batch CICS, IMS, and TSO tasks (CTHREAD) to a single instance.

This is a high-water mark.

Field Name: Q3STHWCT
Subsystem Shared Storage Above 2 GB (DB2 10 or later)

This topic shows detailed information about “Statistics - Subsystem Shared Storage Above 2 GB (DB2 10 or later)”.

Statistics - Subsystem Shared Storage Above 2 GB (DB2 10 or later)

The field labels shown in the following sample layout of “Statistics - Subsystem Shared Storage Above 2 GB (DB2 10 or later)” are described in the following section.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>REAL STORAGE IN USE (MB)</td>
<td>2254.52</td>
</tr>
<tr>
<td>SHARED THREAD AND SYSTEM (MB)</td>
<td>2016.57</td>
</tr>
<tr>
<td>SHARED STACK STORAGE (MB)</td>
<td>237.94</td>
</tr>
<tr>
<td>AVERAGE THREAD FOOTPRINT (MB)</td>
<td>2.33</td>
</tr>
</tbody>
</table>

REAL STORAGE IN USE (MB)

The total amount of real storage in use for 64-bit shared storage. This is recorded at the subsystem level.

Note: This field is available in z/OS 1.10 (and maintenance) or later.

Field Name: S225RLU

SHARED THREAD AND SYSTEM (MB)

The amount of real storage in use for 64-bit shared storage. This does not include shared stack storage. This is recorded at the subsystem level.

Note: This field is available in z/OS 1.10 (and maintenance) or later.

Field Name: S225SSR

SHARED STACK STORAGE (MB)

The amount of real storage in use for 64-bit shared stack storage. This is recorded at the subsystem level.

Note: This field is available in z/OS 1.10 (and maintenance) or later.

Field Name: S225KSR

AVERAGE THREAD FOOTPRINT (MB)

The current average real storage in use for subsystem shared storage of active user threads (allied threads + active and pooled DBATs).

Field Name: S225STFR

AUXILIARY STORAGE IN USE (MB)

The total amount of auxiliary storage in use for 64-bit shared storage. This is recorded at the subsystem level.

Note: This field is available in z/OS 1.10 (and maintenance) or later.

Field Name: S225AXU
Subsystem Shared Storage Above 2 GB

**SHARED THREAD AND SYSTEM (MB)**

The amount of auxiliary storage in use for 64-bit shared storage that does not include shared stack storage. This is recorded at the subsystem level.

**Note:** This field is available in z/OS 1.10 (and maintenance) or later.

**Field Name:** S225SSA

**SHARED STACK STORAGE (MB)**

The amount of auxiliary storage in use for 64-bit shared stack storage. This is recorded at the subsystem level.

**Note:** This field is available in z/OS 1.10 (and maintenance) or later.

**Field Name:** S225KSA
This topic shows detailed information about “Statistics - SQL DCL”.

Statistics - SQL DCL

The field labels shown in the following sample layout of “Statistics - SQL DCL” are described in the following section.

<table>
<thead>
<tr>
<th>SQL DCL</th>
<th>QUANTITY</th>
<th>/SECOND</th>
<th>/THREAD</th>
<th>/COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOCK TABLE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>GRANT</td>
<td>1.00</td>
<td>0.01</td>
<td>N/C</td>
<td>0.03</td>
</tr>
<tr>
<td>REVOKE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>SET HOST VARIABLE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>SET CURRENT SQLID</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>SET CURRENT DEGREE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>SET CURRENT RULES</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>SET CURRENT PATH</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>SET CURRENT PRECISION</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>CONNECT TYPE 1</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>CONNECT TYPE 2</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>RELEASE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>SET CONNECTION</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>ASSOCIATE LOCATORS</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>ALLOCATE CURSOR</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>HOLD LOCATOR</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>FREE LOCATOR</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>TOTAL</td>
<td>1.00</td>
<td>0.01</td>
<td>N/C</td>
<td>0.03</td>
</tr>
</tbody>
</table>

**LOCK TABLE**

The number of LOCK TABLE statements executed.

**Field Name:** QXLOCK

**GRANT**

The number of GRANT statements executed.

**Field Name:** QXGRANT

**REVOKE**

The number of REVOKE statements executed.

**Field Name:** QXREVK

**SET HOST VARIABLE**

The number of SET HOST VARIABLE statements executed. The special register that was retrieved is not tracked.

**Field Name:** QXSETHV

**SET CURRENT SQLID**

The number of SET CURRENT SQLID statements executed.

**Field Name:** QXSETSQL

**SET CURRENT DEGREE**

The number of SET CURRENT DEGREE statements executed.

**Field Name:** QXSETCDG
**SQL DCL**

**SET CURRENT RULES**
- The number of SET CURRENT RULES statements executed.
  - **Field Name**: QXSETCRL

**SET CURRENT PATH**
- The number of SET CURRENT PATH statements executed.
  - **Field Name**: QXSETPTH

**SET CURRENT PRECISION**
- The number of SET CURRENT PRECISION statements executed.
  - **Field Name**: QXSETCPR

**CONNECT TYPE 1**
- The number of CONNECT type 1 statements executed.
  - **Field Name**: QXCON1

**CONNECT TYPE 2**
- The number of CONNECT type 2 statements executed.
  - **Field Name**: QXCON2

**RELEASE**
- The number of RELEASE statements executed.
  - **Field Name**: QXREL

**SET CONNECTION**
- The number of SET CONNECTION statements executed.
  - **Field Name**: QXSETCON

**ASSOCIATE LOCATORS**
- The number of SQL ASSOCIATE LOCATORS statements executed.
  - **Field Name**: QXALOCL

**ALLOCATE CURSOR**
- The number of SQL ALLOCATE CURSOR statements executed.
  - **Field Name**: QXALOCC

**HOLD LOCATOR**
- The number of HOLD LOCATOR statements executed.
  - **Field Name**: QXHLDLOC

**FREE LOCATOR**
- The number of times a FREE LOCATOR statement was issued.
  - **Field Name**: QXFRELOC

**TOTAL**
- The total number of DCL statements executed.
  - The exception field name is SSCDCL.
  - **Field Name**: SSCDCL
This topic shows detailed information about “Statistics - SQL DDL”.

### Statistics - SQL DDL

The field labels shown in the following sample layout of “Statistics - SQL DDL” are described in the following section.

<table>
<thead>
<tr>
<th>SQL DDL</th>
<th>QUANTITY</th>
<th>/SECOND</th>
<th>/THREAD</th>
<th>/COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>CREATE TABLE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>CREATE GLOBAL TEMP TABLE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DECLARE GLOBAL TEMP TABLE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>CREATE AUXILIARY TABLE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>CREATE INDEX</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>CREATE VIEW</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>CREATE SYNONYM</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>CREATE TABLESPACE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>CREATE DATABASE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>CREATE STOGROUP</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>CREATE ALIAS</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>CREATE DISTINCT TYPE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>CREATE FUNCTION</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>CREATE PROCEDURE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>CREATE TRIGGER</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>CREATE SEQUENCE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>CREATE ROLE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>CREATE TRUSTED CONTEXT</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>CREATE MASK / PERMISSION</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>CREATE VARIABLE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>ALTER TABLE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>ALTER INDEX</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>ALTER VIEW</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>ALTER TABLESPACE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>ALTER DATABASE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>ALTER STOGROUP</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>ALTER FUNCTION</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>ALTER PROCEDURE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>ALTER SEQUENCE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>ALTER JAR</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>ALTER TRUSTED CONTEXT</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>ALTER MASK / PERMISSION</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>SQL DDL</th>
<th>QUANTITY</th>
<th>/SECOND</th>
<th>/THREAD</th>
<th>/COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>DROP TABLE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DROP INDEX</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DROP VIEW</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DROP SYNONYM</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DROP TABLESPACE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DROP DATABASE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DROP STOGROUP</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DROP ALIAS</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DROP PACKAGE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DROP DISTINCT TYPE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DROP FUNCTION</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DROP PROCEDURE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DROP TRIGGER</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DROP SEQUENCE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DROP ROLE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DROP TRUSTED CONTEXT</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DROP MASK / PERMISSION</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
</tbody>
</table>
### SQL DDL

<table>
<thead>
<tr>
<th>Statement</th>
<th>Count</th>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DROP VARIABLE</td>
<td>0.00</td>
<td></td>
<td>Number of DROP VARIABLE statements executed.</td>
</tr>
<tr>
<td>RENAME TABLE</td>
<td>0.00</td>
<td></td>
<td>Number of RENAME TABLE statements executed.</td>
</tr>
<tr>
<td>RENAME INDEX</td>
<td>0.00</td>
<td></td>
<td>Number of RENAME INDEX statements executed.</td>
</tr>
<tr>
<td>TRUNCATE TABLE</td>
<td>0.00</td>
<td></td>
<td>Number of TRUNCATE TABLE statements executed.</td>
</tr>
<tr>
<td>COMMENT ON</td>
<td>0.00</td>
<td></td>
<td>Number of COMMENT ON statements executed.</td>
</tr>
<tr>
<td>LABEL ON</td>
<td>0.00</td>
<td></td>
<td>Number of LABEL ON statements executed.</td>
</tr>
<tr>
<td>TOTAL</td>
<td>0.00</td>
<td></td>
<td>Total number of SQL DDL statements executed.</td>
</tr>
</tbody>
</table>

**CREATE TABLE**

The number of CREATE TABLE statements executed.

Field Name: QXCRCTAB

**CREATE GLOBAL TEMP TABLE**

The number of CREATE GLOBAL TEMPORARY TABLE statements executed.

Field Name: QXCRGT

**DECLARE GLOBAL TEMP TABLE**

The number of DECLARE GLOBAL TEMPORARY TABLE statements executed.

Field Name: QXDCGT

**CREATE AUXILIARY TABLE**

The number of CREATE AUXILIARY TABLE statements executed.

Field Name: QXCRATB

**CREATE INDEX**

The number of CREATE INDEX statements executed.

Field Name: QXCRINX

**CREATE VIEW**

The number of CREATE VIEW statements executed.

Field Name: QXDEFFVU

**CREATE SYNONYM**

The number of CREATE SYNONYM statements executed.

Field Name: QXCRSYN

**CREATE TABLESPACE**

The number of CREATE TABLESPACE statements executed.

Field Name: QXCTABS

**CREATE DATABASE**

The number of CREATE DATABASE statements executed.

Field Name: QXCRDAB

**CREATE STOGROUP**

The number of CREATE STOGROUP statements executed.

Field Name: QXCRSTG
CREATE ALIAS
   The number of CREATE ALIAS statements executed.
   Field Name: QXCRALS

CREATE DISTINCT TYPE
   The number of CREATE DISTINCT TYPE statements executed.
   Field Name: QXCDIST

CREATE FUNCTION
   The number of CREATE FUNCTION statements executed.
   Field Name: QXCRUDF

CREATE PROCEDURE
   The number of CREATE PROCEDURE statements issued.
   Field Name: QXCRPRO

CREATE TRIGGER
   The number of CREATE TRIGGER statements executed.
   Field Name: QXCTRIG

CREATE SEQUENCE
   The number of CREATE SEQUENCE statements.
   Field Name: QXCRESEQ

CREATE ROLE
   The number of CREATE ROLE statements executed.
   Field Name: QXCRROL

CREATE TRUSTED CONTEXT
   The number of CREATE TRUSTED CONTEXT statements issued.
   Field Name: QXCRCTX

CREATE MASK / PERMISSION
   The number of CREATE MASK and CREATE PERMISSION statements executed.
   Field Name: QXCREMP

CREATE VARIABLE
   The number of CREATE VARIABLE statements.
   Field Name: QXCRTSV

ALTER TABLE
   The number of ALTER TABLE statements executed.
   Field Name: QXALTTA

ALTER INDEX
   The number of ALTER INDEX statements executed.
   Field Name: QXALTIX

ALTER VIEW
SQL DDL

The number of ALTER VIEW statements issued.
Field Name: QXALT VW

ALTER TABLESPACE
The number of ALTER TABLESPACE statements executed.
Field Name: QXALT TS

ALTER DATABASE
The number of ALTER DATABASE statements executed.
Field Name: QXALDAB

ALTER STOGROUP
The number of ALTER STOGROUP statements executed.
Field Name: QXAL ST

ALTER FUNCTION
The number of ALTER FUNCTION statements executed.
Field Name: QXALUDF

ALTER PROCEDURE
The number of ALTER PROCEDURE statements executed.
Field Name: QXALPRO

ALTER SEQUENCE
The number of ALTER SEQUENCE statements.
Field Name: QXALTSEQ

ALTER JAR
The number of ALTER JAR statements issued.
Field Name: QXALTJR

ALTER TRUSTED CONTEXT
The number of ALTER TRUSTED CONTEXT statements issued.
Field Name: QXALTCTX

ALTER MASK / PERMISSION
The number of ALTER MASK and ALTER PERMISSION statements executed.
Field Name: QXALTMP

DROP TABLE
The number of DROP TABLE statements executed.
Field Name: QXDRPTA

DROP INDEX
The number of DROP INDEX statements executed.
Field Name: QXDRPIX

DROP VIEW
The number of DROP VIEW statements executed.
SQL DDL

Field Name: QXDRPVU

DROP SYNONYM
The number of DROP SYNONYM statements executed.

Field Name: QXDRPSY

DROP TABLESPACE
The number of DROP TABLESPACE statements executed.

Field Name: QXDRPTS

DROP DATABASE
The number of DROP DATABASE statements executed.

Field Name: QXDRPDB

DROP STOGROUP
The number of DROP STOGROUP statements executed.

Field Name: QXDRPST

DROP ALIAS
The number of SQL DROP ALIAS statements executed.

Field Name: QXDRPAL

DROP PACKAGE
The number of SQL DROP PACKAGE statements executed.

Field Name: QXDRPPKG

DROP DISTINCT TYPE
The number of DROP DISTINCT TYPE statements executed.

Field Name: QXDDIST

DROP FUNCTION
The number of DROP FUNCTION statements executed.

Field Name: QXDRPFN

DROP PROCEDURE
The number of DROP PROCEDURE statements executed.

Field Name: QXDRPPR

DROP TRIGGER
The number of DROP TRIGGER statements executed.

Field Name: QXDRPTR

DROP SEQUENCE
The number of DROP SEQUENCE statements.

Field Name: QXDROSEQ

DROP ROLE
The number of DROP ROLE statements issued.

Field Name: QXDRPROL
SQL DDL

DROP TRUSTED CONTEXT
The number of DROP TRUSTED CONTEXT statements issued.
Field Name: QXDRPCTX

DROP MASK / PERMISSION
The number of DROP MASK and DROP PERMISSION statements executed.
Field Name: QXDRPMP

DROP VARIABLE
The number of DROP VARIABLE statements.
Field Name: QXDRPSV

RENAME TABLE
The number of RENAME TABLE statements executed.
Field Name: QXRNTAB

RENAME INDEX
The number of RENAME INDEX statements issued.
Field Name: QXRNIX

TRUNCATE TABLE
The number of TRUNCATE TABLE statements issued.
Field Name: QXTRTBL

COMMENT ON
The number of COMMENT ON statements executed.
Field Name: QXCMTON

LABEL ON
The number of LABEL ON statements executed.
Field Name: QXLABON

TOTAL
The total number of DDL statements executed.
Field Name: SSCDDL

This is an exception field.
This topic shows detailed information about “Statistics - SQL DML”.

**Statistics - SQL DML**

The field labels shown in the following sample layout of “Statistics - SQL DML” are described in the following section.

<table>
<thead>
<tr>
<th>SQL DML</th>
<th>QUANTITY</th>
<th>/SECOND</th>
<th>/THREAD</th>
<th>/COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECT</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>INSERT</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>NUMBER OF ROWS</td>
<td>1551.00</td>
<td>8.53</td>
<td>N/C</td>
<td>47.00</td>
</tr>
<tr>
<td>UPDATE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>NUMBER OF ROWS</td>
<td>28.00</td>
<td>0.15</td>
<td>N/C</td>
<td>0.85</td>
</tr>
<tr>
<td>MERGE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DELETE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>NUMBER OF ROWS</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>PREPARE</td>
<td>16.00</td>
<td>0.09</td>
<td>N/C</td>
<td>0.48</td>
</tr>
<tr>
<td>DESCRIBE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>DESCRIBE TABLE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>OPEN</td>
<td>16.00</td>
<td>0.09</td>
<td>N/C</td>
<td>0.48</td>
</tr>
<tr>
<td>CLOSE</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>FETCH</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>NUMBER OF ROWS</td>
<td>16.00</td>
<td>0.09</td>
<td>N/C</td>
<td>0.48</td>
</tr>
<tr>
<td>TOTAL DML</td>
<td>32.00</td>
<td>0.18</td>
<td>N/C</td>
<td>0.97</td>
</tr>
</tbody>
</table>

**SELECT**

The number of SQL SELECT statements executed.

*Field Name:* QXSELECT

**INSERT**

The number of INSERT statements executed.

*Field Name:* QXINSRT

**INSERT - NUMBER OF ROWS**

The number of rows inserted (DB2 field: QXRWSINSRTD).

*Field Name:* SRWINSRT

**UPDATE**

The number of UPDATE statements executed.

*Field Name:* QXUPDTE

**UPDATE - NUMBER OF ROWS**

The number of rows updated (DB2 field: QXRWSUPDTD).

*Field Name:* SRWUPDAT

**MERGE**

The number of times a MERGE statement was executed.

*Field Name:* QXMERGE

**DELETE**

The number of DELETE statements executed.
**SQL DML**

**Field Name:** QXDELET

**DELETE - NUMBER OF ROWS**

The number of rows deleted (DB2 field: QXRWSDELETED).

**Field Name:** SRWDELETE

**PREPARE**

The number of SQL PREPARE statements executed. This number at the server location might not match the user application because of DDF's internal processing.

**Field Name:** QXPREP

**DESCRIBE**

The number of DESCRIBE, DESCRIBE CURSOR, DESCRIBE INPUT, and DESCRIBE PROCEDURE statements executed. This number at the server location might not match the user application because of DDF's internal processing.

**Field Name:** QXDESC

**DESCRIBE TABLE**

The number of DESCRIBE TABLE statements executed.

**Field Name:** QXDSCTB

**OPEN**

The number of OPEN statements executed.

**Field Name:** QXOPEN

**CLOSE**

The number of CLOSE statements executed. This number at the server location might not match the user application because of DDF's internal processing.

**Field Name:** QXCLOSE

**FETCH**

The number of FETCH statements executed. This number at the server location might not match the user application because of DDF's internal processing.

**Field Name:** QXFETCH

**FETCH - NUMBER OF ROWS**

The number of rows fetched (DB2 field: QXRWSFETCHD).

**Field Name:** SRWFETCH

**TOTAL**

The total number of SQL DML statements executed.

**Field Name:** SSCDML.

This is an exception field.
This topic shows detailed information about “Statistics - Triggers”.

**Statistics - Triggers**

The field labels shown in the following sample layout of “Statistics - Triggers” are described in the following section.

<table>
<thead>
<tr>
<th>TRIGGERS</th>
<th>QUANTITY</th>
<th>/SECOND</th>
<th>/THREAD</th>
<th>/COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>STATEMENT TRIGGER ACTIVATED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>ROW TRIGGER ACTIVATED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
<tr>
<td>SQL ERROR OCCURRED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
<td>0.00</td>
</tr>
</tbody>
</table>

**STATEMENT TRIGGER ACTIVATED**

The number of times a statement trigger was activated.

*Field Name:* QXSTTRG

**ROW TRIGGER ACTIVATED**

The number of times a row trigger was activated.

*Field Name:* QXROWTRG

**SQL ERROR OCCURRED**

The number of times an SQL error occurred during the execution of a triggered action. This includes errors that occur in user-defined functions or stored procedures that are called from triggers and that pass back a negative SQLCODE.

*Field Name:* QXTRGERR
Use Currently Committed

This topic shows detailed information about “Statistics - Use Currently Committed”.

Statistics - Use Currently Committed

The field labels shown in the following sample layout of “Statistics - Use Currently Committed” are described in the following section.

<table>
<thead>
<tr>
<th>USE CURRENTLY COMMITTED</th>
<th>QUANTITY /SECOND</th>
<th>THREAD</th>
<th>COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>INSERT ROWS SKIPPED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
</tr>
<tr>
<td>DELETE ROWS ACCESSED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
</tr>
<tr>
<td>UPDATE ROWS ACCESSED</td>
<td>0.00</td>
<td>0.00</td>
<td>N/C</td>
</tr>
</tbody>
</table>

**INSERT ROWS SKIPPED**

The number of rows skipped by read transactions because of uncommitted INSERT operations (using currently committed semantic for FETCH).

Field Name: QISTRCCI

**DELETE ROWS ACCESSED**

The number of rows accessed by read transactions because of uncommitted DELETE operations (using currently committed semantic for FETCH).

Field Name: QISTRCCD

**UPDATE ROWS ACCESSED**

The number of rows accessed by read transactions because of uncommitted UPDATE operations (using currently committed semantic for FETCH).

Field Name: QISTRCCU
User-Defined Functions

This topic shows detailed information about “Statistics - User-Defined Functions”.

Statistics - User-Defined Functions

The field labels shown in the following sample layout of “Statistics - User-Defined Functions” are described in the following section.

<table>
<thead>
<tr>
<th>USER DEFINED FUNCTIONS</th>
<th>QUANTITY /SECOND /THREAD /COMMIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXECUTED</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
<tr>
<td>ABENDED</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
<tr>
<td>TIMED OUT</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
<tr>
<td>REJECTED</td>
<td>0.00 0.00 N/C 0.00</td>
</tr>
</tbody>
</table>

EXECUTED

The number of user-defined functions executed.

Field Name: QXCAUD

ABENDED

The number of times a user-defined function abended.

Field Name: QXCAUDAB

TIMED OUT

The number of times a user-defined function timed out while waiting to be scheduled.

Field Name: QXCAUDTO

REJECTED

The number of times a user-defined function was rejected.

Field Name: QXCAUDRJ
This topic shows detailed information about “Statistics - Workfile Database”.

This block shows information about the Workfile Database used by DB2 as storage for work files for processing SQL statements, and as storage for created and declared global temporary tables.

The performance metrics in the report block distinguish between work files for declared global temporary tables (DGTTs) and work files for non-DGTT data such as created global temporary tables or sort results. In addition, DB2 supports in-memory work files which are sufficient for performing simple operations and do not require physical allocations. In-memory work files may overflow to physical records in the Workfile Database in case of memory constraints.

Statistics - Workfile Database

The field labels shown in the following sample layout of “Statistics - Workfile Database” are described in the following section.

WORKFILE DATABASE QUANTITY /SECOND /THREAD /COMMIT
--------------------------- -------- ------- ------- -------
TOTAL STORAGE CONFIG (KB) 256.00 N/A N/A N/A
TOT DGTT STOR CONFIG (KB) 128.00 N/A N/A N/A
TOT WF STOR CONFIG (KB) 128.00 N/A N/A N/A
TOTAL STORAGE THRESHOLD (%) 90.00 N/A N/A N/A
MAX TOTAL STORAGE USED (KB) 128.00 N/A N/A N/A
MAX DGTT STOR USED (KB) 64.00 N/A N/A N/A
MAX WF STORAGE USED (KB) 64.00 N/A N/A N/A
CUR TOTAL STORAGE USED (KB) 2.06 N/A N/A N/A
CUR DGTT STOR USED (KB) 1.00 N/A N/A N/A
CUR WF STORAGE USED (KB) 1.06 N/A N/A N/A
STORAGE IN 4K TS (KB) 2.06 N/A N/A N/A
STORAGE IN 32K TS (KB) 64.00 N/A N/A N/A
4K USED INSTEAD OF 32K TS 0.00 0.00 N/C 0.00
32K USED INSTEAD OF 4K TS 0.00 0.00 N/C 0.00
MAX ACTIVE (DM) IN-MEMORY 0.00 N/A N/A N/A
MAX ACT (NONSORT) IN-MEM 0.00 N/A N/A N/A
CUR ACTIVE (DM) IN-MEMORY 0.00 N/A N/A N/A
CUR ACT (NONSORT) IN-MEM 0.00 N/A N/A N/A
MAX STOR (DM) IN-MEM (KB) 0.00 N/A N/A N/A
CUR STOR (DM) IN-MEM (KB) 0.00 N/A N/A N/A
MAX ACTIVE (SORT) IN-MEMORY 0.00 N/A N/A N/A
CUR ACTIVE (SORT) IN-MEMORY 0.00 N/A N/A N/A
MAX STOR (SORT) IN-MEM (KB) 0.00 N/A N/A N/A
CUR STOR (SORT) IN-MEM (KB) 0.00 N/A N/A N/A
IN-MEM (NONSORT) OVERFLOWED 0.00 0.00 N/C 0.00
IN-MEM WORKF NOT CREATED 0.00 0.00 N/C 0.00
AGENT STORAGE CONFIG (KB) 0.00 N/A N/A N/A
NUMBER OF LIMIT EXCEEDED 0.00 0.00 N/C 0.00
AGENT STORAGE THRESHOLD (%) 90.00 N/A N/A N/A
MAX AGENT STORAGE USED (KB) 0.00 N/A N/A N/A

TOTAL STORAGE CONFIG (KB)

The total storage (KB) configured for all table spaces in the Workfile Database.

Field Name: QISTWSTG

TOT DGTT STOR CONFIG (KB)
The total preferred storage (KB) configured for DGTTs in the Workfile Database.

Field Name: QISTDGTTSTG

**TOT WF STOR CONFIG (KB)**

The total preferred storage (KB) configured for non-DGTT work files in the Workfile Database.

Field Name: QISTWFSTG

**TOTAL STORAGE THRESHOLD (%)**

The alert threshold of high space-usage for DGTTs or non-DGTT work files in the Workfile Database (derived from zparm WFSTGUSE_SYSTEM_THRESHOLD).

Field Name: QISTSSTH

**MAX TOTAL STORAGE USED (KB) (DB2 9)**

For DB2 9: The maximum total amount of storage (KB) ever used in the Workfile Database at system level since DB2 startup (DB2 field: QISTWMXU).

Field Name: SISTWFMU

**MAX TOTAL STORAGE USED (KB) (DB2 10)**

The maximum total amount of storage (KB) ever used in the Workfile Database at system level since DB2 startup.

Field Name: QISTWMXU

**MAX DGTT STOR USED (KB)**

The maximum total amount of storage (KB) ever used for DGTTs in the Workfile Database by all agents on the system since DB2 startup.

Field Name: QISTDGTTMXU

**MAX WF STORAGE USED (KB)**

The maximum total amount of storage (KB) ever used for non-DGTT work files in the Workfile Database by all agents on the system since DB2 startup.

Field Name: QISTWFMXU

**CUR TOTAL STORAGE USED (KB) (DB2 9)**

For DB2 9: The total amount of storage (KB) currently used in the Workfile Database at system level.

Field Name: SISTWFCU

**CUR TOTAL STORAGE USED (KB) (DB2 10)**

The total amount of storage (KB) currently used in the Workfile Database at system level.

Field Name: QISTWCTO

**CUR DGTT STOR USED (KB)**

The total amount of storage (KB) currently used for DGTTs in the Workfile Database by all agents on the system.

Field Name: QISTDGTTCTO
Workfile Database

**CUR WF STORAGE USED (KB)**

The total amount of storage (KB) currently used for non-DGTT work files in the Workfile Database by all agents on the system.

Field Name: QISTWFCTO

**STORAGE IN 4K TS (KB)**

For DB2 9: The total amount of storage (KB) currently used for 4 KB table spaces in the Workfile Database.

Field Name: SISTWF04

**STORAGE IN 4K TS (KB)**

The total amount of storage (KB) currently used for 4 KB table spaces in the Workfile Database.

Field Name: QISTW4K

**STORAGE IN 32K TS (KB)**

For DB2 9: The total amount of storage (KB) currently used for 32 KB table spaces in the Workfile Database.

Field Name: SISTWF32

**STORAGE IN 32K TS (KB)**

The total amount of storage (KB) currently used for 32 KB table spaces in the Workfile Database.

Field Name: QISTW32K

**4K USED INSTEAD OF 32K TS**

The number of times that space in a 4 KB page table space was used because space in a 32 KB page table space was preferred but not available in the Workfile Database.

Field Name: QISTWFp2

**32K USED INSTEAD OF 4K TS**

The number of times that space in a 32 KB page table space was used because space in a 4 KB page table space was preferred but not available in the Workfile Database.

Field Name: QISTWFp1

**MAX ACTIVE (DM) IN-MEMORY**

The maximum number of in-memory work files (created by the Data Manager) that were active at any point in time since DB2 startup. This is a high-water mark count.

Field Name: QISTIMAH

**MAX ACT (NONSORT) IN-MEM**

The maximum number of non-SORT related in-memory work files created by the Data Manager that were active at any point in time since DB2 startup. This is a high-water mark count.

Field Name: QISTI2AH

**CUR ACTIVE (DM) IN-MEMORY**
Workfile Database

The number of currently active in-memory work files created by the Data
Manager.

Field Name: QISTIMAC

CUR ACT (NONSORT) IN-MEM

The number of currently active non-SORT related in-memory work files
created by the Data Manager.

Field Name: QISTI2AC

MAX STOR (DM) IN-MEM (KB)

The maximum space used for active in-memory work files created by the
Data Manager at any point in time since DB2 startup. This is a high-water
mark count.

Field Name: QISTIMSH

CUR STOR (DM) IN-MEM (KB)

The total space used for currently active in-memory work files created by
the Data Manager.

Field Name: QISTIMSC

MAX ACTIVE (SORT) IN-MEMORY

The maximum number of in-memory work files created by the SORT
component that were active at any point in time since DB2 start. This is a
high-water mark count.

Field Name: QISTSIAH

CUR ACTIVE (SORT) IN-MEMORY

The number of currently active in-memory work files created by the SORT
component.

Field Name: QISTSIAC

MAX STOR (SORT) IN-MEM (KB)

The maximum space used for active in-memory work files created by the
SORT component at any point in time since DB2 startup. This is a
high-water mark count.

Field Name: QISTSISH

CUR STOR (SORT) IN-MEM (KB)

The total space used for currently active in-memory work files created by
the SORT component.

Field Name: QISTSISC

IN-MEM (NONSORT) OVERFLOWED

The number of times non-SORT related in-memory work files overflowed
into a physical table space.

Field Name: QISTI2OF

IN-MEM WORKF NOT CREATED

The number of times an in-memory work file was not created due to
critical storage conditions.

Field Name: QISTIMNC
Workfile Database

AGENT MAX STORAGE (DB2 9)

For DB2 9: The maximum amount of storage (KB) in the Workfile Database that can be used by each agent (derived from ZPARM MAXTEMPS).

Field Name: SISTWFMX

AGENT STORAGE CONFIG (KB)

The maximum amount of storage (KB) in the Workfile Database that can be used by each agent (derived from ZPARM MAXTEMPS).

Field Name: QISTWMXA

NUMBER OF LIMIT EXCEEDED

The number of times the maximum amount of storage that an agent can use in the Workfile database was exceeded.

Field Name: QISTWFNE

AGENT STORAGE THRESHOLD (%)

The alert threshold of high space-usage for DGTTs or non-DGTT work files in the Workfile Database by an agent (derived from ZPARM WFSTGUSE_AGENT_THRESHOLD).

Field Name: QISTASTH

MAX AGENT STORAGE USED (KB)

The maximum amount of storage (KB) ever used in the Workfile Database by any thread since DB2 startup.

Field Name: QISTAMXU
Chapter 50. The Statistics Save-File Utility

Use the Save-File utility to migrate and convert Statistics Save data sets into a format suitable for OMEGAMON XE for DB2 PE V5.2.

The function performed is specified in a parameter on the EXEC command.

“Migrating Data Sets” on page 50-2
This topic describes how to migrate Statistics Save data sets created by OMEGAMON XE for DB2 PE V4.2, V5.1.0, or V5.1.1 into the record format of OMEGAMON XE for DB2 PE V5.2.

“Converting Data Sets” on page 50-3
To store performance data in the Performance Database you must first convert Statistics Save data sets of OMEGAMON XE for DB2 PE V5.2 into sequential data sets that can be used by the DB2 load utility.

“Save-File Utility DD Statements” on page 50-4
This topic lists the DD statements needed for migration and conversion. All of the DD statements described here are required.
Migrating Data Sets

This topic describes how to migrate Statistics Save data sets created by OMEGAMON XE for DB2 PE V4.2, V5.1.0, or V5.1.1 into the record format of OMEGAMON XE for DB2 PE V5.2.

To migrate Statistics Save data sets:
1. Define an OMEGAMON XE for DB2 PE V5.2 VSAM data set using IDCAMS as output.
2. Use the MIGRATE function of the Save-File utility to migrate the data sets of OMEGAMON XE for DB2 PE V4.2, V5.1.0, or V5.1.1.

The RKO2SAM library provides the sample job DGOPJSMI, which you can modify to suit your installation.

Note: Save data sets from previous versions V4.2, V5.1.0, or V5.1.1 cannot be restored or converted until migrated to OMEGAMON XE for DB2 PE V5.2 format.
Converting Data Sets

To store performance data in the Performance Database you must first convert Statistics Save data sets of OMEGAMON XE for DB2 PE V5.2 into sequential data sets that can be used by the DB2 load utility.

You can convert data sets in one of the following ways to store performance data in the Performance Database:

- You can use the SAVE subcommand with the CONVERT option to convert and save reduced data into a sequential data set that can be loaded into DB2 tables. The output of this subcommand option is a sequential data set, that is specified and requested in SYSIN. The data set attributes are:

  Organization
  PS

  Record format
  VB

  Record length
  9072

  Block size
  9076

For more information refer to [Reporting User's Guide](#).  

- You can use the converted Save-File data sets to generate CSV (comma-separated value) input-data. This CSV data can then be transferred to workstations and imported into spreadsheets to improve DB2 performance analysis using graphical representations or pivot tables. For more information refer to [Reporting User's Guide](#).  

- You can use the CONVERT function to convert Statistics Save data sets of OMEGAMON XE for DB2 PE V5.2 into sequential data sets that can be used by the DB2 load utility. This enables you to store performance data in the Performance Database.

The RKO2SAMP library provides the sample job DGOPJSCO, which converts Save data sets into sequential Save-File data sets, suitable for use with the DB2 load utility. You can modify this sample to suit your installation.

The following list shows the types of records that are created by the CONVERT function and where to find their layout descriptions in the sample library RKO2SAMP:

- General data records (DGOSDGEN)
- Buffer pool data records (DGOSDBUF)
- DDF data records (DGOSDDDF)
- Group buffer pool records (DGOSDGBP)
- Buffer pool data set records (DGOSDSET)
- Accelerator data records (DGOSDXCL)
Save-File Utility DD Statements

This topic lists the DD statements needed for migration and conversion. All of the DD statements described here are required.

**Input**

The DDNAME of the input data set. This can be an OMEGAMON XE for DB2 PE V4.2.0, V5.1.0, or V5.1.1 Statistics Save data set for the MIGRATE function, or an OMEGAMON XE for DB2 PE V5.2 Statistics VSAM Save data set for the CONVERT function.

**Output**

The DDNAME of the output data set.

For CONVERT, allocate the data set with the following characteristics:

- **RECFM**: VB
- **LRECL**: 9072
- **BLKSIZE**: 9076

Refer to Chapter 64, “OMEGAMON XE for DB2 PE VSAM Data Sets,” on page 64-1 for details on how to specify the allocated data sets to migrate to OMEGAMON XE for DB2 PE V5.2.

**DPMLOG**

OMEGAMON XE for DB2 PE command processor messages and messages indicating exceptional processing conditions are written to DPMLOG. If DPMLOG is not specified, it is dynamically allocated to the SYSOUT message class of the job. Allocate the data set with the following attributes:

- **RECFM**: FBA
- **LRECL**: 133
- **BLKSIZE**: 6251
Chapter 51. The Statistics File Data Set and Output Records

Use the FILE subcommand to format DB2 Statistics records and write them to sequential data sets suitable for use by the DB2 load utility. You can store unreduced Statistics data into the performance database. Use the performance database to produce tailored reports using a reporting facility such as Query Management Facility (QMF).

The format of the output data from the Statistics File data set is identical with that of the CONVERT function of the Save-File utility.

You can also use the File data sets to generate CSV (comma-separated value) input-data. This CSV data can then be transferred to workstations and imported into spreadsheets to improve DB2 performance analysis using graphical representations or pivot tables. For more information refer to Reporting User’s Guide.

The Statistics File data set is produced when OMEGAMON XE for DB2 PE Statistics delta records are externalized using the FILE subcommand. Each such delta record represents the period of time between two pairs of DB2 Statistics delta records and can be represented in the File data set by up to 3 types of records:

- **General Statistics** records contain data from IFCID 001 and 002. One general Statistics record is produced for each Statistics delta record.

- **Buffer pool Statistics** records contain data derived from IFCID 002 records. One buffer pool record is produced for each buffer pool active at the start time of the corresponding delta record. Each OMEGAMON XE for DB2 PE Statistics delta record can produce up to 80 buffer pool Statistics records.

- **DDF Statistics** records contain DDF Statistics originating from IFCID 001. One DDF record is produced for each remote location communicating with the local DB2 subsystem using DB2 private protocol (as at the start of the corresponding delta record period). Another DDF record is produced for all remote locations that used DRDA (where at least one location used this method at the start time of the delta record). Each OMEGAMON XE for DB2 PE Statistics delta record can produce several DDF Statistics records.

- **Group buffer pool Statistics** records contain data derived from IFCID 002 records. One group buffer pool record is produced for each group buffer pool active at the start time of the corresponding delta record. Each OMEGAMON XE for DB2 PE Statistics delta record can produce up to 80 group buffer pool Statistics records.

- **Buffer pool data set** records contain data derived from IFCID 199. One row is written for each open data set that has an I/O event rate at least one event per second during the reporting interval.

- **Accelerator** records contain data derived from IFCID 002. One row is written for each active accelerator attached to the DB2 subsystem that is currently reported.

File data is written to a File data set. The following types of records are created:

- General data records
- Buffer pool data records
- DDF data records
- Group buffer pool records
- Buffer pool data set records
- Accelerator records

Descriptions of the layouts of these records can be found in the RKO2SAMP library under the following names:

**DGOSDGEN**
General data records

**DGOSDBUF**
Buffer pool data records

**DGOSDDDF**
DDF data records

**DGOSDGBP**
Group buffer pool records

**DGOSDSET**
Buffer pool data set records

**DGOSDXCL**
Accelerator records
Part 10. The System Parameters Report Set

These topics provide information about the system parameters reports.

Note: For an introduction to the System Parameters report set and general system parameter information refer to the Reporting User’s Guide.

- Chapter 52, “System Parameters Report Header,” on page 52-1
  This section introduces the System Parameters report header.
- Chapter 53, “Example of the System Parameters Report,” on page 53-1
  This section shows an example of the System Parameters report.
- Chapter 54, “System Parameters Report Blocks,” on page 54-1
  This section describes the blocks and fields shown in the system parameters report.
- Chapter 55, “Alter Buffer Pool Command Issued,” on page 55-1
  This topic shows detailed information about “System Parameters - Alter Buffer Pool Command Issued”.
- Chapter 56, “Alter Group Buffer Pool Command Issued,” on page 56-1
  This topic shows detailed information about “System Parameters - Alter Group Buffer Pool Command Issued”.
- Chapter 57, “Buffer Pool Parameters,” on page 57-1
  This topic shows detailed information about “System Parameters - Buffer Pool Parameters”.
- Chapter 58, “Group Buffer Pool Parameters,” on page 58-1
  This topic shows detailed information about “System Parameters - Group Buffer Pool Parameters”.

© Copyright IBM Corp. 1985, 2013
Chapter 52. System Parameters Report Header

This section introduces the System Parameters report header.

There are two different types of report headers for system parameters, for:

- MEMBER scope reports.
- GROUP scope reports.

System Parameters Report Header for MEMBER-Scope and GROUP-Scope Reports

Here is an example of a System Parameters report header for MEMBER-scope reports:

LOCATION: STLEC1 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-1
GROUP: GRPA SYSTEM PARAMETERS REPORT
MEMBER: M2
SUBSYSTEM: SSDQ ACTUAL FROM: 01/30/10 22:50:03.98
DB2 VERSION: V10

Here is an example of a System Parameters report header for GROUP-scope reports:

LOCATION: SYS1DSN2 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 4-1
GROUP: DSN2 SYSTEM PARAMETERS REPORT
DB2 VERSION: V10 ACTUAL FROM: 01/30/10 06:10:23.14

Field description

The OMEGAMON XE for DB2 PE system parameters report header contains the following information, described line by line:

LOCATION
The DB2 reporting location. If the location name is not available, the DB2 data sharing group name is printed in this field. If the DB2 data sharing group name does not exist, the DB2 subsystem ID is printed.

OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (VnRnMn)
The product name and the version, release, and modification level.

PAGE
The page number in the format lll-nnnnnn, where lll denotes the location number within the report and nnnnnn the page number within the location.

GROUP
The name of the DB2 data sharing group. This field shows N/A if there is no group name.

SYSTEM PARAMETERS REPORT
The title of the report.

MEMBER
The name of the DB2 data sharing member or the member name of the DB2 subsystem. This field shows N/A if there is no member name.

This field is not printed on the report page showing the GROUP BUFFERPOOLS PARAMETERS.
System Parameters- Report Header

**SUBSYSTEM**
The ID of the DB2 subsystem that generated the data. This field is not printed on the report page showing the GROUP BUFFERPOOLS PARAMETERS.

**ACTUAL FROM/TO**
The date and time of the first and last record included in the log for a location, group, subsystem, or member.

**DB2 VERSION**
The DB2 version number of the subsystem that generated the data.
|

|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|
|

Chapter 53. Example of the System Parameters Report
This section shows an example of the System Parameters report.
LOCATION:
GROUP:
MEMBER:
SUBSYSTEM:
DB2 VERSION:

PMODBE1
DBE1
SE11
SE11
V11

OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0)
SYSTEM PARAMETERS REPORT

ACTUAL FROM: 07/15/13 03:46:00.00

MVS PARMLIB UPDATE PARAMETERS (DSNTIPM)
--------------------------------------SUBSYSTEM DEFAULT (SSID)...................................SE11
SUPPRESS SOFT ERRORS (SUPERRS)..............................YES
STORAGE SIZES INSTALLATION PARMS (DSNTIPC,DSNTIPE,DSNTIPE1)
--------------------------------------------------------------MAX NO OF DATA SETS CONCURRENTLY IN USE (DSMAX)..........20,000
EDM STATEMENT CACHE SIZE IN KB (EDMSTMTC)...............113,386
EDM DBD CACHE SIZE IN KB (EDMDBDC)......................102,400
EDM SKELETON POOL SIZE IN KB (EDM_SKELETON_POOL)........102,400
MAXIMUM SIZE OF EDM POOL IN BYTES (EDMPOOL)...................0
MAXIMUM SIZE OF SORT POOL IN BYTES (SRTPOOL).........10,240,000
MAX IN-MEMORY SORT SIZE (MAXSORT_IN_MEMORY)...............1,000
MAXIMUM SIZE OF RID POOL IN KB (MAXRBLK)................400,000
MAX NO OF USERS CONCURRENTLY RUNNING IN DB2 (CTHREAD).......400
MAX NO OF CONCURRENT REMOTE ACTIVE CONNECTIONS (MAXDBAT)....200
MAX NO OF REMOTE CONNECTIONS (CONDBAT)...................10,000
MAX NO OF TSO CONNECTIONS (IDFORE)..........................200
MAX NO OF BATCH CONNECTIONS (IDBACK)........................200
MAXIMUM KEPT DYNAMIC STATEMENTS (MAXKEEPD)................5,000
MAX OPEN FILE REFS (MAXOFILR)...............................100
CONTRACT THREAD STORAGE (CONTSTOR)..........................YES
MANAGE THREAD STORAGE (MINSTOR)..............................NO
LONG-RUNNING READER IN MINUTES (LRDRTHLD)....................10
DDL TIMEOUT FACTOR (DDLTOX)...................................1
INDEX CLEANUP THREADS (INDEX_CLEANUP_THREADS)................10
TRACING, CHECKPOINT & PSEUDO-CLOSE PARAMETERS (DSNTIPN)
------------------------------------------------------START AUDIT TRACE (AUDITST)..................................NO
START GLOBAL TRACE (TRACSTR).................................NO
TRACE TABLE SIZE IN 4K BYTES (TRACTBL).......................16
START SMF ACCOUNTING (SMFACCT)................................1
START SMF STATISTICS (SMFSTAT)........................1,3,4,5,6
STATISTICS TIME INTERVAL IN MINUTES (STATIME).................1
SYNCHRONIZATION INTERVAL WITHIN THE HOUR (SYNCVAL)...........NO
ONLINE DATASET STATISTICS TIME INTERVAL IN MIN.(DSSTIME)......5
START MONITOR TRACE (MON).....................................1
MONITOR BUFFER SIZE IN BYTES (MONSIZE)................1,048,576
UNICODE IFCIDS (UIFCIDS)....................................YES
DDF/RRSAF ACCUM (ACCUMACC)...................................10
AGGREGATION FIELDS (ACCUMUID).................................0
COMPRESS SMF RECS (SMFCOMP).................................OFF
LOCATION:
GROUP:
MEMBER:
SUBSYSTEM:
DB2 VERSION:

PMODBE1
DBE1
SE11
SE11
V11

IRLM INSTALLATION PARAMETERS (DSNTIPI)
-------------------------------------IRLM SUBSYSTEM NAME (IRLMSID)..............................IE11
IRLM RESOURCE TIMEOUT IN SECONDS (IRLMRWT)...................30
IRLM AUTOMATIC START (IRLMAUT)..............................YES
IRLM START PROCEDURE NAME (IRLMPRC)....................SE11IRLM
SECONDS DB2 WILL WAIT FOR IRLM START (IRLMSWT)..............120
U LOCK FOR REPEATABLE READ OR READ STABILITY (RRULOCK)......YES
X LOCK FOR SEARCHED UPDATE/DELETE (XLKUPDLT).................NO
IMS/BMP TIMEOUT FACTOR (BMPTOUT)..............................4
IMS/DLI TIMEOUT FACTOR (DLITOUT)..............................6
WAIT FOR RETAINED LOCKS (RETLWAIT)............................0
ENABLE DB CHECKING...........................................NO
IRLM PROCESSING PARAMETERS
-------------------------WAIT TIME FOR LOCAL DEADLOCK..............................5,000
NUMBER OF LOCAL CYCLES PER GLOBAL CYCLE.......................1
TIMEOUT INTERVAL.............................................30
IRLM MAXIMUM CSA USAGE ALLOWED................................0
Z/OS LOCK TABLE HASH ENTRIES..........................1,048,576
PENDING NUMBER OF HASH ENTRIES................................0
Z/OS LOCK TABLE LIST ENTRIES..............................8,282
MAX 31-BIT IRLM PRIVATE STORAGE...............................0
MAX 64-BIT IRLM PRIVATE STORAGE...............................0
ARCHIVE LOG INSTALLATION PARAMETERS (DSNTIPA)
--------------------------------------------CATALOG ARCHIVE DATASETS (CATALOG)..........................YES
COPY1 ARCHIVE LOG DEVICE TYPE (UNIT).......................DASD
COPY2 ARCHIVE LOG DEVICE TYPE (UNIT2)...................’BLANK’
SPACE ALLOCATION METHOD (ALCUNIT)......................CYLINDER
PRIMARY SPACE ALLOCATION (PRIQTY)...........................100
SECONDARY SPACE ALLOCATION (SECQTY)..........................10
ARCHIVE LOG BLOCK SIZE IN BYTES (BLKSIZE)................24,576
MAXIMUM READ TAPE UNITS (MAXRTU)..............................2
TAPE UNIT DEALLOCATION PERIOD (DEALLCT).................0000:00
MAX NUMBER OF DATASETS RECORDED IN BSDS (MAXARCH)........10,000
FIRST ARCHIVE COPY MASS STG GROUP NAME...................’NONE’
SECOND ARCHIVE COPY MASS STG GROUP NAME..................’NONE’
DAYS TO RETAIN ARCHIVE LOG DATA SETS (ARCRETN)...............30
ISSUE WTOR BEFORE MOUNT FOR ARCHIVE VOLUME (ARCWTOR)........YES
COMPACT DATA (COMPACT).......................................NO
QUIESCE PERIOD (QUIESCE)......................................5
SINGLE VOLUME (SVOLARC)......................................NO

OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0)
SYSTEM PARAMETERS REPORT

DISTRIBUTED DATA FACILITY PANEL 2 (DSNTIP5)
------------------------------------------TCP/IP ALREADY VERIFIED (TCPALVER)...........................NO
EXTRA BLOCKS REQ (EXTRAREQ).................................100
EXTRA BLOCKS SRV (EXTRASRV).................................100
TCP/IP KEEPALIVE (TCPKPALV).................................120
CONNECTION QUEUE MAX DEPTH (MAXCONQN).........................0
CONNECTION QUEUE MAX WAIT (MAXCONQW)..........................0
POOL THREAD TIMEOUT (POOLINAC)..............................120
PROTECTION INSTALLATION PARAMETERS (DSNTIPP)
-------------------------------------------ARCHIVE LOG RACF PROTECTION (PROTECT)........................NO
DB2 AUTHORIZATION ENABLED (AUTH)............................YES
PLAN AUTHORIZATION CACHE SIZE (AUTHCACH)..................3,072
PACKAGE AUTHORIZATION CACHE SIZE (CACHEPAC)...........5,242,880
ROUTINE AUTHORIZATION CACHE SIZE (CACHERAC)...........5,242,880
AUTH EXIT CHECK (AUTHEXIT_CHECK)........................PRIMARY
AUTH EXIT CACHE REFRESH (AUTHEXIT_CACHEREFRESH)............NONE
SYSTEM ADMINISTRATOR 1 AUTHORIZATION ID (SYSADM)...........HELM
SYSTEM ADMINISTRATOR 2 AUTHORIZATION ID (SYSADM2)........SYSADM
SYSTEM OPERATOR 1 AUTHORIZATION ID (SYSOPR1)...............HELM
SYSTEM OPERATOR 2 AUTHORIZATION ID (SYSOPR2)...............EMIL

© Copyright IBM Corp. 1985, 2013

PAGE: 1-1

PAGE: 1-2
ACTUAL FROM: 07/15/13 03:46:00.00

DEFINE GROUP OR MEMBER (DSNTIPK)
-------------------------------GROUP NAME (GRPNAME).......................................DBE1
MEMBER NAME (MEMBNAME).....................................SE11
DEL CF STRUCTS (DEL_CFSTRUCTS_ON_RESTART)....................NO
DISTRIBUTED DATA FACILITY PANEL 1 (DSNTIPR)
------------------------------------------DDF STARTUP OPTION (DDF)...................................AUTO
RLST ACCESS ERROR (RLFERRD).............................NOLIMIT
RESYNCHRONIZATION INTERVAL IN MINUTES (RESYNC)................2
DBAT STATUS (CMTSTAT)..................................INACTIVE
IDLE THREAD TIMEOUT INTERVAL (IDTHTOIN).....................120
EXTENDED SECURITY (EXTSEC)..................................YES
MAX TYPE 1 INACTIVE THREADS (MAXTYPE1)........................0
LOCK ESCALATION PARAMETERS (DSNTIPJ)
-----------------------------------MAX PAGE OR ROW LOCKS PER TABLE SPACE (NUMLKTS)...........2,000
MAX PAGE OR ROW LOCKS PER USER (NUMLKUS).................10,000
LOG INSTALLATION PARAMETERS (DSNTIPL,DSNTIPH)
---------------------------------------------

53-1


## System Parameters - Report

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DB2 Version Install (DSNTPA1)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Install DD Control (RGFINSTL)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Data Sharing Enabled (DSHARE)</strong></td>
<td>YES</td>
</tr>
<tr>
<td><strong>DB2 Version (DSNTPA1)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Application Programming Defaults Panel 1 (DSNTPP)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Default Hand Language (DEFLANG)</strong></td>
<td>IBMCOB</td>
</tr>
<tr>
<td><strong>Minimum Divide Scale (DECDIV)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Decimal Arithmetic (DECARTH)</strong></td>
<td>15</td>
</tr>
<tr>
<td><strong>Default DML (DML)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Default DML (DML)</strong></td>
<td>DEFAULT</td>
</tr>
<tr>
<td><strong>Default SQL Delimiter (SQLDELL1)</strong></td>
<td>APPOST</td>
</tr>
<tr>
<td><strong>Default MIXED GRAPHIC (MIXED)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>EBCDIC SBCS CSSID (SCCSID)</strong></td>
<td>1,148</td>
</tr>
<tr>
<td><strong>ASCII SBCS CSSID (ASCSSID)</strong></td>
<td>019</td>
</tr>
<tr>
<td><strong>EBCDIC MBCS CSSID (MCSSID)</strong></td>
<td>N/P</td>
</tr>
<tr>
<td><strong>ASCII MBCS CSSID (ASCSSID)</strong></td>
<td>N/P</td>
</tr>
<tr>
<td>** Unicode SBCS CSSID (USCSSID)**</td>
<td>1,208</td>
</tr>
<tr>
<td><strong>Default Encoding Scheme (ENSHEME)</strong></td>
<td>EBCDIC</td>
</tr>
<tr>
<td><strong>Application Encoding (APPRCH)</strong></td>
<td>EBCDIC</td>
</tr>
<tr>
<td><strong>Locale LC_Type (LC_TYPE)</strong></td>
<td>EBCDIC</td>
</tr>
<tr>
<td><strong>Default Decfloat Round Mode (DEF_DECFLOAT_ROUND_MODE)</strong></td>
<td>ROUND_HALF_EVEN</td>
</tr>
<tr>
<td><strong>SQL Object Defaults Panel (DSNTPP,DSNTPP1)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Object Create Format (OBJECT_CREATE_FORMAT)</strong></td>
<td>EXTENDED</td>
</tr>
<tr>
<td><strong>Utility Object Conversion (UTILITY_OBJECT_CONVERSION)</strong></td>
<td>NONE</td>
</tr>
<tr>
<td><strong>Vary OS Control Interval (OSVICI)</strong></td>
<td>YES</td>
</tr>
<tr>
<td><strong>Table Space Allocation in KB (TOSQTY)</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>Index Space Allocation in KB (IXQTY)</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>Optimize Extent Sizing (MUEXTXZ)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Pad Index By Default (PADIA)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Default Partition Ssegsize (DPSSGSS2)</strong></td>
<td>32</td>
</tr>
<tr>
<td><strong>Percent Free For Update (PCTFREE_UPD)</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>Define Data Sets (IMPODEF)</strong></td>
<td>YES</td>
</tr>
<tr>
<td><strong>Use Data Compression (IMPSMCP)</strong></td>
<td>YES</td>
</tr>
<tr>
<td><strong>Limit Key Conv Part Tab (IX_TB_PART_CONV_EXCLUDE)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Min Performance (PERFMON)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Cache Dynamic SQL (CACHEDYN)</strong></td>
<td>YES</td>
</tr>
<tr>
<td><strong>Optimization Hints Allowed (OPTHINTS)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Skip Uncmm Inserts (SKIPUNCI)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Immediate Override Flag (IMPREQI)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Indexed Plan Default (INDEXMNDE)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Rand Plan Default (RPLANMNDE)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Plan Default (PLANMNDE)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Max Degree of Parallelism (PARAMDEG)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Buffer Pool Parameters (DSNTPP)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Default 4-KB Buffer Pool for User Data (TSBPPOOL)</strong></td>
<td>BP2</td>
</tr>
<tr>
<td><strong>Evaluate Uncommitted (EVALUNCOM)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Current Refresh Age (RESHAGE)</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>Current Max Storage (MAXTEMPS)</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>Location</strong>: <strong>PMODE1</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Member</strong>: <strong>SE11</strong></td>
<td></td>
</tr>
</tbody>
</table>

**System Parameters - Report**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DB2 Version (VII)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Application Programming Defaults Panel 2 (DSNTPP2,DSNTPP21)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Default Hand Language (DEFLANG)</strong></td>
<td>IBMCOB</td>
</tr>
<tr>
<td><strong>Minimum Divide Scale (DECDIV)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Decimal Arithmetic (DECARTH)</strong></td>
<td>15</td>
</tr>
<tr>
<td><strong>Default DML (DML)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Default DML (DML)</strong></td>
<td>DEFAULT</td>
</tr>
<tr>
<td><strong>Default SQL Delimiter (SQLDELL1)</strong></td>
<td>APPOST</td>
</tr>
<tr>
<td><strong>Default MIXED GRAPHIC (MIXED)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>EBCDIC SBCS CSSID (SCCSID)</strong></td>
<td>1,148</td>
</tr>
<tr>
<td><strong>ASCII SBCS CSSID (ASCSSID)</strong></td>
<td>019</td>
</tr>
<tr>
<td><strong>EBCDIC MBCS CSSID (MCSSID)</strong></td>
<td>N/P</td>
</tr>
<tr>
<td><strong>ASCII MBCS CSSID (ASCSSID)</strong></td>
<td>N/P</td>
</tr>
<tr>
<td>** Unicode SBCS CSSID (USCSSID)**</td>
<td>1,208</td>
</tr>
<tr>
<td><strong>Default Encoding Scheme (ENSHEME)</strong></td>
<td>EBCDIC</td>
</tr>
<tr>
<td><strong>Application Encoding (APPRCH)</strong></td>
<td>EBCDIC</td>
</tr>
<tr>
<td><strong>Locale LC_Type (LC_TYPE)</strong></td>
<td>EBCDIC</td>
</tr>
<tr>
<td><strong>Default Decfloat Round Mode (DEF_DECFLOAT_ROUND_MODE)</strong></td>
<td>ROUND_HALF_EVEN</td>
</tr>
<tr>
<td><strong>SQL Object Defaults Panel (DSNTPP,DSNTPP1)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Object Create Format (OBJECT_CREATE_FORMAT)</strong></td>
<td>EXTENDED</td>
</tr>
<tr>
<td><strong>Utility Object Conversion (UTILITY_OBJECT_CONVERSION)</strong></td>
<td>NONE</td>
</tr>
<tr>
<td><strong>Vary OS Control Interval (OSVICI)</strong></td>
<td>YES</td>
</tr>
<tr>
<td><strong>Table Space Allocation in KB (TOSQTY)</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>Index Space Allocation in KB (IXQTY)</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>Optimize Extent Sizing (MUEXTXZ)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Pad Index By Default (PADIA)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Default Partition Ssegsize (DPSSGSS2)</strong></td>
<td>32</td>
</tr>
<tr>
<td><strong>Percent Free For Update (PCTFREE_UPD)</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>Define Data Sets (IMPODEF)</strong></td>
<td>YES</td>
</tr>
<tr>
<td><strong>Use Data Compression (IMPSMCP)</strong></td>
<td>YES</td>
</tr>
<tr>
<td><strong>Limit Key Conv Part Tab (IX_TB_PART_CONV_EXCLUDE)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Min Performance (PERFMON)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Cache Dynamic SQL (CACHEDYN)</strong></td>
<td>YES</td>
</tr>
<tr>
<td><strong>Optimization Hints Allowed (OPTHINTS)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Skip Uncmm Inserts (SKIPUNCI)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Immediate Override Flag (IMPREQI)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Indexed Plan Default (INDEXMNDE)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Indexed Plan Default (INDEXMNDE)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Rand Plan Default (RPLANMNDE)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Rand Plan Default (RPLANMNDE)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Default 4-KB Buffer Pool for User Data (TSBPPOOL)</strong></td>
<td>BP2</td>
</tr>
<tr>
<td><strong>Evaluate Uncommitted (EVALUNCOM)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Current Refresh Age (RESHAGE)</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>Current Max Storage (MAXTEMPS)</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>Location</strong>: <strong>PMODE1</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Member</strong>: <strong>SE11</strong></td>
<td></td>
</tr>
</tbody>
</table>

**Database Version (VII)**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Max Degree of Parallelism (PARAMDEG)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Star Join Enabling (STARJOIN)</strong></td>
<td>DISABLE</td>
</tr>
<tr>
<td><strong>Max Data Caching in MB (MAXCACHE)</strong></td>
<td>20</td>
</tr>
<tr>
<td><strong>Current Refresh Age (RESHAGE)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Current Main Type (MAINTYPE)</strong></td>
<td>SYSTEM</td>
</tr>
<tr>
<td><strong>Buffer Pool Parameters (DSNTPP)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Default 4-KB Buffer Pool for User Data (TSBPPOOL)</strong></td>
<td>BP2</td>
</tr>
<tr>
<td><strong>Evaluate Uncommitted (EVALUNCOM)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Current Query Acceler (QUERY_ACCELERATION)</strong></td>
<td>NO</td>
</tr>
<tr>
<td><strong>Workfile Database Panel (DSNTPP1)</strong></td>
<td></td>
</tr>
</tbody>
</table>

**Group**: **SE11**

**Location**: **PMODE1**

**Member**: **SE11**

---

OMEGAMON XE for DB2 Performance Expert (Y2RM2)

**Page**: 1-3

**Actual From**: 07/15/13 03:46:00.00
System Parameters- Report

Max Temp RID (MaxTemps_RID).................................NOLIMIT
Agent Level Threshold (Wfsgue.Agent Threshold)...............0
System Level Threshold (Wfsgue_System_THRESHOLD)...........90

Location: Pmodbe1  Omgamon Xe For Db2 Performance Expert (Vsr2m0)  Page: 1-5
Group: DBE1  System Parameters Report
Member: SE11
Subsystem: SE11  Actual from: 07/15/13 03:46:00.00
Db2 Version: V11

Other System Parameters

- Dual BSDS Mode (words): YES
- Roll Up Parallel Task Accounting (ptaskroll): YES
- No. Pages Small Table Threshold (nputhresh): 0
- Offload Option (offload): YES
- Su Conversion Factor: .281
- Minimum Divide Scale (mindivsc): NONE
- Star Join Threshold (stables): 10
- Online System Park User Id Monitor: N/P
- Online System Park Core Id Monitor: N/P
- Online System Park Time Changed: N/P
- Online System Park Type: N/P
- Db2-Supplied Deqp Indicator: N/P
- Max Concurrent Pkg Ops (MaxConcurrentPkgOps): 10
- Admin Scheduler Jcl Proc Name (AdmJcProc): N/P
- Free Local Cached Statements (CachedynFreelocal): 0
- Index I/O Parallelism (IndexIOParallelism): YES
- Zoometrics: YES
- Use Tracked For Implicit Ts (ImpTracked): YES
- Dszie For Implicit Ts (ImposedSize): 4
- Enable Multiple Index Access (Subo_Ridx): YES
- Spt01 Inline Length (Spt01InlineLength): 32,136
- Ddf Compatibility (DdfCompatibility): NO
- Dyn Stmt Cache Stor (CacheDdpTrackStorLim): 0

Db2 Utilities Parameters (Dsntip6, Dsntip61, Dsntip62)

- Default Startup Modules (Dsntip03)
  - Parameter Module (Dsnzparm)
  - Access Control (Access_Cntl_Module)
  - Identify/Auth (Idauth_Module)
  - Signon (Signon_Module)
  - Security Administrator 1 Type (Secadm1_Type)

Db2-Supplied Deqp Indicator: N/P

Temporary Unit Name (Voltdrv): SYSDA
Statistics History (StatHist): NONE
Statistics Rollup (Statroll): YES
Utility Timeout Factor (utimout): UTC

Template Time (Template_Time): UTC
Max Degree Of Utility Parallelism (Paramdeg_util): 99
System-Level Backups (SystemLevelBackups): NO
Restore/Recover (Restore_Recover_fromXmp): NO
Dump Class Name (Utils_Dump_Class_Name): "Blank"
Max Tape Units (Restore_TapeUnits): 0
Default Template (Ccppopor)
Db1.Lob.... jedoch au.
Redeq Part Sort Nps1 (Redeq_Part_sort_Nps1): AUTO
Redeq List Processing (Redeq_List_Processing): PARALLEL
Redeq Mapping Database (Redeq.Mapping Database): N/P
Redeq Drop Pkg Parts (Redeq_Drop_Pkg_Parts): NO
Redeq Ignore Free Space (Redeq Ignore FreeSpace): NO

Location: Pmodbe1  Omgamon Xe For Db2 Performance Expert (Vsr2m0)  Page: 1-6
Group: DBE1  System Parameters Report
Member: SE11
Subsystem: SE11  Actual from: 07/15/13 03:46:00.00
Db2 Version: V11

Buffer Pool Parameters

Timestamp 07/15/13 03:46:00.00
Buffer Pool 1D  Bp1  Vpool Size (Pages): 5000
Vpool Seq Thresh: 80
Horiz Defere White Thresh: 30
Vert Defere White Thresh (%): 5
Vert Defere White Thresh (Buf): 0
Vpool Parallel Seq Thresh: 50
Assisting Parallel Seq Thresh: 0
PfFix Attribute: NO
Page Steal Method: LRU
Autosize: NO
FrameSize: 4K
Vpool Size MIn: 0
Vpool Size Max: 0

Timestamp 07/15/13 03:46:00.00
Buffer Pool 1D  Bp1  Vpool Size (Pages): 10000
Vpool Seq Thresh: 80
Horiz Defere White Thresh: 30
Vert Defere White Thresh (%): 5
Vert Defere White Thresh (Buf): 0
Vpool Parallel Seq Thresh: 50
Assisting Parallel Seq Thresh: 0
PfFix Attribute: NO
Page Steal Method: LRU
Autosize: NO
FrameSize: 4K
Vpool Size MIn: 0
Vpool Size Max: 0

Chapter 53. Example of the System Parameters Report 53-3
## System Parameters - Report

<table>
<thead>
<tr>
<th>Buffer Pool ID</th>
<th>VPOOL Size (Pages)</th>
<th>Seq Thresh</th>
<th>Horiz Def Thresh</th>
<th>Vert Def Thresh (%)</th>
<th>Vert Def Thresh (Buf)</th>
<th>VPOOL Parallel Seq Thresh</th>
<th>Assist Seq Thresh</th>
<th>pgfix attribute</th>
<th>Page Steal Method</th>
<th>Autosize</th>
<th>Frame Size</th>
<th>VPOOL Size Min</th>
<th>VPOOL Size Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>BP2</td>
<td>20000</td>
<td>80</td>
<td>30</td>
<td>5</td>
<td>0</td>
<td>50</td>
<td>0</td>
<td>NO</td>
<td>LRU</td>
<td>NO</td>
<td>4K</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>BP3</td>
<td>5000</td>
<td>80</td>
<td>30</td>
<td>5</td>
<td>0</td>
<td>50</td>
<td>0</td>
<td>NO</td>
<td>LRU</td>
<td>NO</td>
<td>4K</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>BP7</td>
<td>10000</td>
<td>80</td>
<td>30</td>
<td>5</td>
<td>0</td>
<td>50</td>
<td>0</td>
<td>NO</td>
<td>LRU</td>
<td>NO</td>
<td>4K</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>BP9</td>
<td>10</td>
<td>80</td>
<td>30</td>
<td>5</td>
<td>0</td>
<td>50</td>
<td>0</td>
<td>NO</td>
<td>LRU</td>
<td>NO</td>
<td>4K</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>BP10</td>
<td>35</td>
<td>80</td>
<td>30</td>
<td>5</td>
<td>0</td>
<td>50</td>
<td>0</td>
<td>NO</td>
<td>LRU</td>
<td>NO</td>
<td>4K</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>BP11</td>
<td>35</td>
<td>80</td>
<td>30</td>
<td>5</td>
<td>0</td>
<td>50</td>
<td>0</td>
<td>NO</td>
<td>LRU</td>
<td>NO</td>
<td>4K</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>BUFFER POOL ID</td>
<td>TIMESTAMP</td>
<td>VPOOL SIZE (PAGES)</td>
<td>VPOOL SEQ THRESH</td>
<td>HORIZ DEFER WRITE THRESH</td>
<td>VERT DEFER WRITE THRESH (%)</td>
<td>VERT DEFER WRITE THRESH (BUF)</td>
<td>VPOOL PARALLEL SEQ THRESH</td>
<td>ASSISTING PARALLEL SEQ THRESH</td>
<td>PGFIX ATTRIBUTE</td>
<td>PAGE STEAL METHOD</td>
<td>AUTOSIZE</td>
<td>FRAMESIZE</td>
<td>VPOOL SIZE MIN</td>
</tr>
<tr>
<td>---------------</td>
<td>------------------------</td>
<td>--------------------</td>
<td>------------------</td>
<td>--------------------------</td>
<td>-----------------------------</td>
<td>-------------------------------</td>
<td>-----------------------------</td>
<td>-------------------------------</td>
<td>----------------</td>
<td>----------------</td>
<td>----------</td>
<td>-----------</td>
<td>--------------</td>
</tr>
<tr>
<td>BP12</td>
<td>07/15/13 03:46:00.00</td>
<td>50</td>
<td>80</td>
<td>30</td>
<td>5</td>
<td>0</td>
<td>50</td>
<td>0</td>
<td>0</td>
<td>LRU</td>
<td>No</td>
<td>4K</td>
<td>0</td>
</tr>
<tr>
<td>BP13</td>
<td>07/15/13 03:46:00.00</td>
<td>50</td>
<td>80</td>
<td>30</td>
<td>5</td>
<td>0</td>
<td>50</td>
<td>0</td>
<td>0</td>
<td>LRU</td>
<td>No</td>
<td>4K</td>
<td>0</td>
</tr>
<tr>
<td>BP14</td>
<td>07/15/13 03:46:00.00</td>
<td>50</td>
<td>80</td>
<td>30</td>
<td>5</td>
<td>0</td>
<td>50</td>
<td>0</td>
<td>0</td>
<td>LRU</td>
<td>No</td>
<td>4K</td>
<td>0</td>
</tr>
<tr>
<td>BP15</td>
<td>07/15/13 03:46:00.00</td>
<td>75</td>
<td>80</td>
<td>30</td>
<td>5</td>
<td>0</td>
<td>50</td>
<td>0</td>
<td>0</td>
<td>LRU</td>
<td>No</td>
<td>4K</td>
<td>0</td>
</tr>
<tr>
<td>BP32K</td>
<td>07/15/13 03:46:00.00</td>
<td>1000</td>
<td>80</td>
<td>30</td>
<td>5</td>
<td>0</td>
<td>50</td>
<td>0</td>
<td>0</td>
<td>LRU</td>
<td>No</td>
<td>4K</td>
<td>0</td>
</tr>
</tbody>
</table>
## System Parameters - Report

<table>
<thead>
<tr>
<th>Buffer Pool ID</th>
<th>Timestamp</th>
<th>Pool Size (Pages)</th>
<th>Seq Thresh</th>
<th>Horiz Def Thresh</th>
<th>Vert Def Thresh (%)</th>
<th>Vert Def Thresh (Buf)</th>
<th>Pool Parallel Seq Thresh</th>
<th>Assisting Parallel Seq Thresh</th>
<th>Pgfix Attribute</th>
<th>Page Steal Method</th>
<th>AutoSize</th>
<th>FrameSize</th>
<th>VPool Size Min</th>
<th>VPool Size Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>BP32K1</td>
<td>07/15/13 03:46:00.00</td>
<td>5000</td>
<td>80</td>
<td>30</td>
<td>5</td>
<td>0</td>
<td>50</td>
<td>0</td>
<td>No</td>
<td>LRU</td>
<td>No</td>
<td>4K</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>BP8K1</td>
<td>07/15/13 03:46:00.00</td>
<td>10000</td>
<td>80</td>
<td>30</td>
<td>5</td>
<td>0</td>
<td>50</td>
<td>0</td>
<td>No</td>
<td>LRU</td>
<td>No</td>
<td>4K</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>BP8K4</td>
<td>07/15/13 03:46:00.00</td>
<td>2000</td>
<td>80</td>
<td>30</td>
<td>5</td>
<td>0</td>
<td>50</td>
<td>0</td>
<td>No</td>
<td>LRU</td>
<td>No</td>
<td>4K</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>BP16K0</td>
<td>07/15/13 03:46:00.00</td>
<td>2000</td>
<td>80</td>
<td>30</td>
<td>5</td>
<td>0</td>
<td>50</td>
<td>0</td>
<td>No</td>
<td>LRU</td>
<td>No</td>
<td>4K</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
### Buffer Pool Parameters

<table>
<thead>
<tr>
<th>TimeStamp</th>
<th>Buffer Pool ID</th>
<th>Buffer Pool Size (Pages)</th>
<th>Vpool Seq Thresh</th>
<th>Horiz Defer Write Thresh</th>
<th>Vert Defer Write Thresh (%)</th>
<th>Vert Defer Write Thresh (Buf)</th>
<th>Vpool Parallel Seq Thresh</th>
<th>Assisting Parallel Seq Thresh</th>
<th>Pgfix Attribute</th>
<th>Page Steal Method</th>
<th>AutoSize</th>
<th>FrameSize</th>
<th>Vpool Size Min</th>
<th>Vpool Size Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>07/15/13 03:46:00.00</td>
<td>BP16K1</td>
<td>2000</td>
<td>80</td>
<td>30</td>
<td>5</td>
<td>0</td>
<td>50</td>
<td>0</td>
<td>NO</td>
<td>LRU</td>
<td>NO</td>
<td>4K</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

### Group Buffer Pool Parameters

<table>
<thead>
<tr>
<th>TimeStamp</th>
<th>Member</th>
<th>Allocated GBP Size (4K)</th>
<th>GBP Class Castout Threshold (%)</th>
<th>GBP Class Castout Threshold (Pages)</th>
<th>Actual Directory</th>
<th>Actual Data Entry</th>
<th>GBP Checkpoint Interval (Min)</th>
<th>Pending Directory to Data Ratio</th>
<th>Pending Directory to Data Ratio (Auto Rec)</th>
<th>Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>07/15/13 03:46:00.01</td>
<td>SE11</td>
<td>768</td>
<td>5</td>
<td>0</td>
<td>1414</td>
<td>282</td>
<td>4</td>
<td>5</td>
<td>YES</td>
<td>SIMPLEX</td>
</tr>
<tr>
<td>07/15/13 03:46:00.01</td>
<td>SE11</td>
<td>1536</td>
<td>5</td>
<td>0</td>
<td>231</td>
<td>44</td>
<td>4</td>
<td>5</td>
<td>YES</td>
<td>SIMPLEX</td>
</tr>
<tr>
<td>07/15/13 03:46:00.01</td>
<td>SE11</td>
<td>1280</td>
<td>5</td>
<td>0</td>
<td>330</td>
<td>5</td>
<td>4</td>
<td>5</td>
<td>YES</td>
<td>SIMPLEX</td>
</tr>
</tbody>
</table>

### System Parameters Report

Chapter 53. Example of the System Parameters Report
Chapter 54. System Parameters Report Blocks

This section describes the blocks and fields shown in the system parameters report.

Blocks are listed in alphabetical order, fields are shown in the order they appear in the block.

"Application Programming Defaults Panel 1 (DSNTIPF)" on page 54-3
This topic shows detailed information about “System Parameters - Application Programming Defaults Panel 1 (DSNTIPF)”.

"Application Programming Defaults Panel 2 (DSNTIP4, DSNTIP41)” on page 54-9
This topic shows detailed information about “System Parameters - Application Programming Defaults Panel 2 (DSNTIP4, DSNTIP41)”.

"Archive Log Installation Parameters (DSNTIPA)” on page 54-13
This topic shows detailed information about “System Parameters - Archive Log Installation Parameters (DSNTIPA)”.

"Buffer Pool Parameters (DSNTIP1)” on page 54-17
This topic shows detailed information about “System Parameters - Buffer Pool Parameters (DSNTIP1)”.

"Data Definition Control Support (DSNTIPZ)” on page 54-19
This topic shows detailed information about “System Parameters - Data Definition Control Support (DSNTIPZ)”.

"Define Group or Member (DSNTIPK)” on page 54-22
This topic shows detailed information about “System Parameters - Define Group or Member (DSNTIPK)”.

"Databases and Spaces Started Automatically (DSNTIPS)” on page 54-24
This topic shows detailed information about “System Parameters - Databases and Spaces Started Automatically (DSNTIPS)”.

"Default Startup Modules (DSNTIPO3)” on page 54-25
This topic shows detailed information about “System Parameters - Default Startup Modules (DSNTIPO3)”.

"Distributed Data Facility Panel 1 (DSNTIPR)” on page 54-26
This topic shows detailed information about “System Parameters - Distributed Data Facility Panel 1 (DSNTIPR)”.

"Distributed Data Facility Panel 2 (DSNTIP5)” on page 54-29
This topic shows detailed information about “System Parameters - Distributed Data Facility Panel 2 (DSNTIP5)”.

"DB2 Utilities Parameters (DSNTIP6, DSNTIP61, DSNTIP62)” on page 54-32
This topic shows detailed information about “System Parameters - DB2 Utilities Parameters (DSNTIP6, DSNTIP61, DSNTIP62)”.

"DB2 Version Install (DSNTIPA1)” on page 54-36
This topic shows detailed information about “System Parameters - DB2 Version Install (DSNTIPA1)”.

"IRLM Installation Parameters (DSNTIPI)” on page 54-37
This topic shows detailed information about “System Parameters - IRLM Installation Parameters (DSNTIPI)”.

"IRLM Processing Parameters” on page 54-40
This topic shows detailed information about “System Parameters - IRLM Processing Parameters”.
This topic shows detailed information about “System Parameters - Lock Escalation Parameters (DSNTIPJ)”. This topic shows detailed information about “System Parameters - Log Installation Parameters (DSNTIPL, DSNTIPH)”. This topic shows detailed information about “System Parameters - List of Long Names”. This topic shows detailed information about “System Parameters - MVS Parmlib Update Parameters (DSNTIPM)”. This topic shows detailed information about “System Parameters - Operator Functions Installation Parameters (DSNTIPO)”. This topic shows detailed information about “System Parameters - Other System Parameters”. This topic shows detailed information about “System Parameters - Performance and Optimization (DSNTIP8, DSNTIP81)”. This topic shows detailed information about “System Parameters - Protection Installation Parameters (DSNTIPP)”. This topic shows detailed information about “System Parameters - Protection Panel (DSNTIPP1)”. This topic shows detailed information about “System Parameters - Query Accelerator Preferences (DSNTIP82)”. This topic shows detailed information about “SystemParameters - Routine Parameters (DSNTIPX)”. This topic shows detailed information about “System Parameters - Sizes Panel 1 (DSNTIPD)”. This topic shows detailed information about “System Parameters - SQL Object Defaults Panel (DSNTIP7, DSNTIP71)”. This topic shows detailed information about “System Parameters - Storage Sizes Installation Parm (DSNTIPC, DSNTIPE, DSNTIPE1)”. This topic shows detailed information about “System Parameters - Tracing, Checkpoint & Pseudo-Close Parameters (DSNTIPN)”. This topic shows detailed information about “System Parameters - Workfile Database Panel (DSNTIP91)”. This topic shows detailed information about “System Parameters - List of Long Names”. This topic shows detailed information about “System Parameters - MVS Parmlib Update Parameters (DSNTIPM)”. This topic shows detailed information about “System Parameters - Operator Functions Installation Parameters (DSNTIPO)”. This topic shows detailed information about “System Parameters - Other System Parameters”. This topic shows detailed information about “System Parameters - Performance and Optimization (DSNTIP8, DSNTIP81)”. This topic shows detailed information about “System Parameters - Protection Installation Parameters (DSNTIPP)”. This topic shows detailed information about “System Parameters - Protection Panel (DSNTIPP1)”. This topic shows detailed information about “System Parameters - Query Accelerator Preferences (DSNTIP82)”. This topic shows detailed information about “SystemParameters - Routine Parameters (DSNTIPX)”. This topic shows detailed information about “System Parameters - Sizes Panel 1 (DSNTIPD)”. This topic shows detailed information about “System Parameters - SQL Object Defaults Panel (DSNTIP7, DSNTIP71)”. This topic shows detailed information about “System Parameters - Storage Sizes Installation Parm (DSNTIPC, DSNTIPE, DSNTIPE1)”. This topic shows detailed information about “System Parameters - Tracing, Checkpoint & Pseudo-Close Parameters (DSNTIPN)”. This topic shows detailed information about “System Parameters - Workfile Database Panel (DSNTIP91)". This topic shows detailed information about “System Parameters - List of Long Names”. This topic shows detailed information about “System Parameters - MVS Parmlib Update Parameters (DSNTIPM)”. This topic shows detailed information about “System Parameters - Operator Functions Installation Parameters (DSNTIPO)”. This topic shows detailed information about “System Parameters - Other System Parameters”. This topic shows detailed information about “System Parameters - Performance and Optimization (DSNTIP8, DSNTIP81)”. This topic shows detailed information about “System Parameters - Protection Installation Parameters (DSNTIPP)”. This topic shows detailed information about “System Parameters - Protection Panel (DSNTIPP1)”. This topic shows detailed information about “System Parameters - Query Accelerator Preferences (DSNTIP82)”. This topic shows detailed information about “SystemParameters - Routine Parameters (DSNTIPX)”. This topic shows detailed information about “System Parameters - Sizes Panel 1 (DSNTIPD)”. This topic shows detailed information about “System Parameters - SQL Object Defaults Panel (DSNTIP7, DSNTIP71)”. This topic shows detailed information about “System Parameters - Storage Sizes Installation Parm (DSNTIPC, DSNTIPE, DSNTIPE1)”. This topic shows detailed information about “System Parameters - Tracing, Checkpoint & Pseudo-Close Parameters (DSNTIPN)”. This topic shows detailed information about “System Parameters - Workfile Database Panel (DSNTIP91)".
Application Programming Defaults Panel 1 (DSNTIPF)

This topic shows detailed information about “System Parameters - Application Programming Defaults Panel 1 (DSNTIPF)”.

This block shows application programming defaults.

The values shown are used as default values by the program preparation panels, program preparation CLIST (DSNH), and precompiler. They can also be used as defaults by other programs, such as Query Management Facility (QMF).

Changing some of these defaults is not recommended because changes can make the syntax of existing SQL statements invalid or affect the way application programs run.

Values set here are contained in load module DSNHDECP, in library prefix.SDSNEXIT, which can be loaded and accessed by application programs. When modifying DSNHDECP, do so only by changing and running the installation CLIST.

Do not modify the data in DSNHDECP. If you modify any installation parameters by changing job DSNTIJUZ directly, these values are not recorded for later updates, new installations, or migrations.

Note: The fields shown on this panel depend on the installed DB2 version.

System Parameters - Application Programming Defaults Panel 1 (DSNTIPF)

The field labels shown in the following sample layout of “System Parameters - Application Programming Defaults Panel 1 (DSNTIPF)” are described in the following section.

APPLICATION PROGRAMMING DEFAULTS PANEL 1 (DSNTIPF)
--------------------------------------------------
DEFAULT HOST LANGUAGE (DEFLANG).......................IBMCOB
DECIMAL POINT OPTION (DECIMAL)............................PERIOD
DEFAULT DELIMITER (DELM)...............................DEFAULT
DEFAULT SQL DELIMITER (SQLDELI)...........................DEFAULT
DIST SQL STRING DELIMITER (DSQLDELI).....................APOST
DEFAULT MIXED GRAPHIC (MIXED)................................NO
EBCDIC SBCS CCSID (SCCSID)..................................1,148
EBCDIC MBCS CCSID (MCCSID)..................................N/P
EBCDIC GBCS CCSID (GCCSID)..................................N/P
ASCII SBCS CCSID (ASCCSID).............................819
ASCII MBCS CCSID (AMCCSID).............................N/P
ASCII GBCS CCSID (AGCCSID).............................N/P
UNICODE SBCS CCSID (USCCSID)............................367
UNICODE MBCS CCSID (UMCCSID)............................1,208
UNICODE GBCS CCSID (UGCCSID)............................1,200
DEFAULT ENCODING SCHEME (ENSCHEME).....................EBCDIC
APPLICATION ENCODING (APPENSCHE)......................EBCDIC
LOCALE LC_TYPE (LC_CTYPE).............................'BLANK'
DECIMAL ROUND MODE (DEF_DECIMAL_ROUND_MODE)........ROUND_HALF_EVEN
DEFAULT CHARSET (CHARSET)..............................ALPHANUM

DEFAULT HOST LANGUAGE (DEFLANG)

The default programming language for your site. This can be:
- ASM
- C
Application Programming Defaults Panel 1 (DSNTIPF)

- CPP
- COBOL
- COB2
- IBMCOB
- FORTRAN
- PLI

When this is C or C++, you can fold SQL identifiers to uppercase.

Install parameter LANGUAGE DEFAULT on panel DSNTIPF, or ZPARM DEFLANG in DSNHDECP.

**Field Name:** QWPBLANG

**DECIMAL POINT OPTION (DECIMAL)**

Indicates whether the decimal contains a comma (,) or a period (.). This parameter is used for dynamic SQL and COBOL programs. It is not used or supported by other languages.

Install parameter DECIMAL POINT IS on panel DSNTIPF, or ZPARM DECIMAL in DSNHDECP.

*Derivation:* DB2 field QWPBDE

**Field Name:** QWPBDE

**DEFAULT DELIMITER (DELIM)**

Shows the string delimiter for COBOL. Default string delimiter is the quotation mark. This option is applicable to all types of COBOL.

Install parameter STRING DELIMITER on panel DSNTIPF, or ZPARM DELIM in DSNHDECP.

**Field Name:** QWPBDL

**DEFAULT SQL DELIMITER (SQLDELI)**

The string delimiter for SQL.

Install parameter SQL STRING DELIMITER on panel DSNTIPF, or ZPARM SQLDELI in DSNHDECP.

*Derivation:* DB2 field QWPBS SDL

**Field Name:** QWPBS SDL

**DIST SQL STRING DELIMITER (DSQLDELI)**

Shows the SQL string delimiter used by this DB2 for bind operations when the requester does not give DB2 that information.

Install parameter DIST SQL STR DELIMTR on panel DSNTIPF, or ZPARM DSQLDELI in DSNHDECP.

**Field Name:** QWPBDS SD

**DEFAULT MIXED GRAPHIC (MIXED)**

Indicates whether the code points X'0E' and X'0F' are the shift-out and shift-in controls for character strings that include double-byte characters.

Install parameter MIXED DATA on panel DSNTIPF, or ZPARM MIXED in DSNHDECP.

**Field Name:** QWPBGRA
EBCDIC SBCS CCSID (SCCSID)

The EBCDIC single-byte coded character set ID.

A coded character set identifier (CCSID) must be specified when DDF STARTUP OPTION field on panel DSNTIPR is set to AUTO or COMMAND, or when the MIXED DATA field on panel DSNTIPF is set to YES. When mixed data is used, valid Mixed Data CCSID must also be specified.

A nonexistent CCSID causes an error.
An incorrect CCSID can corrupt data.

Install parameter EBCDIC CCSID on panel DSNTIPF, or ZPARM SCCSID in DSNHDECP.

Field Name: QWPBSID

EBCDIC MBCS CCSID (MCCSID)

The EBCDIC mixed coded character set ID.

A coded character set identifier (CCSID) must be specified when DDF STARTUP OPTION field on panel DSNTIPR is set to AUTO or COMMAND, or when the MIXED DATA field on panel DSNTIPF is set to YES. When mixed data is used, valid Mixed Data CCSID must also be specified.

A nonexistent CCSID causes an error.
An incorrect CCSID can corrupt data.

Install parameter EBCDIC CCSID on panel DSNTIPF, or ZPARM MCCSID in DSNHDECP.

Field Name: QWPBMID

EBCDIC GBCS CCSID (GCCSID)

The EBCDIC graphic coded character set ID.

A coded character set identifier (CCSID) must be specified when DDF STARTUP OPTION field on panel DSNTIPR is set to AUTO or COMMAND, or when the MIXED DATA field on panel DSNTIPF is set to YES. When mixed data is used, valid Mixed Data CCSID must also be specified.

A nonexistent CCSID causes an error.
An incorrect CCSID can corrupt data.

Install parameter EBCDIC CCSID on panel DSNTIPF, or ZPARM GCCSID in DSNHDECP.

Field Name: QWPBGID

ASCII SBCS CCSID (ASCSID)

The ASCII single-byte coded character set ID.

The default (0) means the installation has no ASCII databases, table spaces, or tables.

Install parameter ASCII CCSID on panel DSNTIPF, or ZPARM ASCCSID in DSNHDECP.

Field Name: QWPBASID
ASCII MBCS CCSID (AMCCSID)
Indicates the ASCII mixed coded character set ID.
The default (0) means the installation has no ASCII databases, table spaces, or tables.
Install parameter ASCII CCSID on panel DSNTIPF, or ZPARM AMCCSID in DSNHDECP.
Field Name: QWPBAMID

ASCII GBCS CCSID (AGCCSID)
Indicates the ASCII graphic coded character set ID.
The default (0) means the installation has no ASCII databases, table spaces, or tables.
Install parameter ASCII CCSID on panel DSNTIPF, or ZPARM AGCCSID in DSNHDECP.
Field Name: QWPBAGID

UNICODE SBCS CCSID (USCCSID)
Unicode Single Byte Character Set identification.
Parameter UNICODE CCSID in installation panel DSNTIPF, or ZPARM USCCSID in macro DSNHDECP.
Field Name: QWPBUSID

UNICODE MBCS CCSID (UMCCSID)
Unicode Mixed Character Set identification.
Parameter UNICODE CCSID in installation panel DSNTIPF, or ZPARM UMCCSID in macro DSNHDECP.
Field Name: QWPBUMID

UNICODE GBCS CCSID (UGCCSID)
Unicode graphics character set identification.
Parameter UNICODE CCSID in installation panel DSNTIPF, or ZPARM UGCCSID in macro DSNHDECP.
Field Name: QWPBUGID

DEFAULT ENCODING SCHEME (ENSCHEME)
The default encoding scheme, which can be ASCII or EBCDIC, or UNICODE.
Install parameter DEF ENCODING SCHEME on panel DSNTIPF, or ZPARM ENSCHEME in DSNHDECP.
Derivation: DB2 field QWPBENS
Field Name: QWPBENS

APPLICATION ENCODING (APPENSCH)
Application encoding scheme.
Install parameter APPLICATION ENCODING on installation panel DSNTIPF, or ZPARM APPENSCH in DSNHDECP.
Field Name: QWPBAPSC
LOCALE LC_CTYPE (LC_TYPE)

The system LOCALE LC_CTYPE.

A locale is the part of the system environment that depends on language and cultural conventions. An LC_TYPE is a subset of a locale that applies to character functions. The UPPER, LOWER, and TRANSLATE scalar functions use the CURRENT LOCALE LC_CTYPE system default or special register. The results of these functions can vary, depending on the setting of the locale.

The following values are possible:

BLANK

The source field is empty.

This is the default, unless it is necessary to run the UPPER, LOWER, or TRANSLATE functions for data that must be interpreted using the rules provided by specific locales, for example, En_US or Fr_CA.

1st word

The source field contains left-justified word(s), where each byte of a word is > X'40'. It can be a single word or several ones, delimited by bytes <= X'40'.

Note: These hexadecimal codes do not represent printable characters.

N/P

The source field contains regular words that are not left-justified. This means that the first bytes are <= X'40'. N/P is also shown if the whole source field only consists of bytes < X'40', such as zeros.

Install parameter LOCALE LC_CTYPE on panel DSNTIPF, or ZPARM LC_TYPE in DSNHDECP.

Field Name: QWPBLCTP

DECFLOAT ROUND MODE (DEF_DECFLOAT_ROUND_MODE)

The default rounding mode for the decimal floating point type. Possible values are:

X'80'  ROUND_CEILING
X'40'  ROUND_DOWN
X'20'  ROUND_FLOOR
X'10'  ROUND_HALF_DOWN
X'08'  ROUND_HALF_EVEN
X'04'  ROUND_HALF_UP
X'02'  ROUND_UP

Otherwise this field shows 'BLANK'.

ZPARM DEF_DECFLOAT_ROUND_MODE in DSNHDECP.

Field Name: QWPBDDRM

DEFAULT CHARSET (CHARSET)

Shows the default character set, ALPHANUM or KATAKANA.
Application Programming Defaults Panel 1 (DSNTIPF)

ZPARM CHARSET in DSNHDECN.

Field Name: QWPBCHAR
This topic shows detailed information about “System Parameters - Application Programming Defaults Panel 2 (DSNTIP4, DSNTIP41)”. This block is a continuation of DSNTIPF and shows application programming defaults.

The values shown are used as default values by the program preparation panels, the program preparation CLIST (DSNH), and the precompiler. They can also be used as defaults by other programs, such as Query Management Facility (QMF).

Changing some of these defaults is not recommended because changes can make the syntax of existing SQL statements invalid or affect the way application programs run.

**System Parameters - Application Programming Defaults Panel 2 (DSNTIP4, DSNTIP41)**

The field labels shown in the following sample layout of “System Parameters - Application Programming Defaults Panel 2 (DSNTIP4, DSNTIP41)” are described in the following section.

**APPLICATION PROGRAMMING DEFAULTS PANEL 2 (DSNTIP4, DSNTIP41)**

```plaintext
MINIMUM DIVIDE SCALE (DECDIV3)...............................NO
DECIMAL ARITHMETIC (DECARTH).................................15
USE FOR DYNAMIC RULES (DYNRULS).............................YES
STATIC DESCRIBE (DESCSTAT)..................................YES
DATE FORMAT (DATE)..........................................ISO
TIME FORMAT (TIME)..........................................ISO
LOCAL DATE LENGTH (DATELEN).................................N/A
LOCAL TIME LENGTH (TIMELEN).................................N/A
IMPLICIT TIMEZONE.......................................CURRENT
STD SQL LANGUAGE (STDSQL)....................................NO
PAD NULL-TERMINATED (PADNTSTR)...............................NO
APPL COMPAT LEVEL (APPLCOMPAT)............................V1R1
LIKE BLANK INSIGNIFICANT (LIKE_BLANK_INSIGNIFICANT)........NO
```

**MINIMUM DIVIDE SCALE (DECDIV3)**

This field is for IBM service use.

*Field Name:* QWP4DIV3

**DECIMAL ARITHMETIC (DECARTH)**

Indicates the rules of precision for a decimal field.

Install parameter DECIMAL ARITHMETIC on panel DSNTIP4, or ZPARM DECARTH in DSNHDECP.

*Derivation:* DB2 field QWPBAR

*Field Name:* QWPBARTH

**USE FOR DYNAMIC RULES (DYNRULS)**

Shows whether DB2 uses the application programming defaults specified on this panel or those of the DB2 precompiler options for dynamic SQL statements bound using DYNAMICRULES bind, define, or invoke behavior.
Application Programming Defaults Panel 2 (DSNTIP4, DSNTIP41)

When YES, the application programming (DSNHDECP) defaults are used for dynamic SQL statements in plans or packages bound using DYNAMICRULES bind, define, or invoke behavior.

The following defaults are affected:

- DECIMAL POINT IS
- STRING DELIMITER
- SQL STRING DELIMITER
- MIXED DATA
- DECIMAL ARITHMETIC

When NO, values of the precompiler options are used for dynamic SQL statements in plans or packages bound with DYNAMICRULES(BIND).

Install parameter USE FOR DYNAMICRULES on panel DSNTIP4, or ZPARM DYNRULES in DSNHDECP.

Field Name: QWPBDRLS

STATIC DESCRIBE (DESCSTAT)

Shows whether DB2 builds a DESCRIBE SQLDA when binding static SQL statements.

A DESCRIBE cannot be issued against a static SQL statement except:

- In a distributed environment, where DB2 for z/OS is the server and the requester supports extended dynamic SQL. In this instance, a DESCRIBE on an SQL statement in the extended dynamic package appears to DB2 as a DESCRIBE on a static SQL statement in the DB2 package.
- When an application uses a stored procedure result set, the application must allocate a cursor for that result set. The application can do this using a DESCRIBE CURSOR statement. The SQL statement actually described is the one with the cursor declared in the stored procedure. If that statement is static, a static SQL statement must be described.

When NO (default), DB2 does not generate a DESCRIBE SQLDA at BIND time for static SQL statements. If a DESCRIBE request is received at execution time, DB2 generates an error. However, if the describe request comes from a DESCRIBE CURSOR statement, DB2 satisfies the request but is only able to provide data type and length information. Column names are not provided.

When YES, DB2 generates a DESCRIBE SQLDA at BIND time so that DESCRIBE requests for static SQL can be satisfied during execution.

Note: You must rebind packages after this value has been set to YES.

This option increases the size of some packages because the DESCRIBE SQLDA is now stored with each statically-bound SQL SELECT statement.

Install parameter DESCRIBE FOR STATIC on panel DSNTIP4, or ZPARM DESCSTAT in DSN6SPRM.

Field Name: QWP4DSST

DATE FORMAT (DATE)

Default output format for dates.

Valid formats are ISO (yyyy-mm-dd), USA (mm/dd/yyyy), EUR (dd.mm.yyyy), JIS (yyyy- mm-dd), or LOCAL (your choice, defined by a
date exit routine). DB2 interprets the input date from the punctuation and converts the output date to the required format.

Install parameter DATE FORMAT on panel DSNTIP4, or ZPARM DATE in DSNHDECP.

Field Name: QWPBDATE

TIME FORMAT (TIME)

Indicates the default output format for times.

Valid values are ISO (hh:mm:ss), USA (hh:mm AM), EUR (hh:mm:ss), JIS (hh:mm:ss), or LOCAL (your choice, defined by a time exit routine). DB2 interprets the input time from the punctuation and converts the output time to the required format.

Install parameter TIME FORMAT on panel DSNTIP4, or ZPARM TIME in DSNHDECP.

Field Name: QWPBTIME

LOCAL DATE LENGTH (DATELEN)

Shows the length of the longest field required to hold a locally defined date.

The default (0) indicates an IBM-supplied format (ISO, JIS, USA, or EUR).

Install parameter LOCAL DATE LENGTH on panel DSNTIP4, or ZPARM DATELEN in DSNHDECP.

Field Name: QWPBDLEN

LOCAL TIME LENGTH (TIMELEN)

Shows the length of the longest field required to hold a time when a locally defined time format is used.

The default (0) indicates an IBM-supplied format (ISO, JIS, USA, or EUR).

Install parameter LOCAL TIME LENGTH on panel DSNTIP4, or ZPARM TIMELEN in DSNHDECP.

Field Name: QWPBTLEN

IMPLICIT TIMEZONE

The implicit time zone that is associated with DB2 table columns and routine parameters that are declared as time stamp with time zone.

For IFCID 106 - Application Programming Defaults, this field is displayed twice, with its hex value and in a readable string.

This field corresponds to DSNHDECP field IMPLICIT_TIMEZONE.

Field Name: QWPBIMTZ

STD SQL LANGUAGE (STDSQL)

Shows whether SQL, the language standard used by applications, conforms to 1986 ANSI SQL standard.

YES Conforms to the 1986 ANSI SQL standard
NO Conforms to the SQL language defined by DB2
86 Conforms to the 1986 ANSI SQL standard
Application Programming Defaults Panel 2 (DSNTIP4, DSNTIP41)

Install parameter STD SQL LANGUAGE on panel DSNTIP4, or ZPARM STDSQL in DSNHDECP.

Field Name: QWPBSQL

PAD NULL-TERMINATED (PADNTSTR)

Shows whether output host variables that are NULL-terminated strings are padded with blanks and a NULL terminator.

When NO, NULL-terminated output host variables have the NULL terminator placed at the end of actual data returned in the host variable.
When YES, NULL-terminated output host variables have the NULL terminator placed at the end of the string, after the string has been padded with blanks from the end of the actual data to the declared length of the output host variable.

Install parameter PAD NUL-TERMINATED on installation panel DSNTIP4, or ZPARM PADNTSTR in DSNHDECP.

Field Name: QWPBPAD

APPL COMPAT LEVEL (APPLCOMPAT)

Specifies the DB2 level for downward compatibility with applications. The ZPARM name is APPLCOMPAT in DSN6SPRM.

Field Name: QWP4APCO_VAR

LIKE BLANK INSIGNIFICANT (LIKE_BLANK_INSIGNIFICANT)

YES indicates that blanks are not significant when DB2 applies the LIKE predicate to a string. Blanks are significant in DB2 10 or 9.

This setting corresponds to field LIKE BLANK INSIGNIFICANT on installation panel DSNTIP41. The ZPARM name is LIKE_BLANK_INSIGNIFICANT in DSN6SPRM.

Field Name: QWP4LBIN
Archive Log Installation Parameters (DSNTIPA)

This topic shows detailed information about “System Parameters - Archive Log Installation Parameters (DSNTIPA)”. This block shows the characteristics of archive log data sets.

System Parameters - Archive Log Installation Parameters (DSNTIPA)

The field labels shown in the following sample layout of “System Parameters - Archive Log Installation Parameters (DSNTIPA)” are described in the following section.

ARCHIVE LOG INSTALLATION PARAMETERS (DSNTIPA)

<table>
<thead>
<tr>
<th>Field Label</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CATALOG ARCHIVE DATASETS (CATALOG)</td>
<td>YES</td>
</tr>
<tr>
<td>COPY1 ARCHIVE LOG DEVICE TYPE (UNIT)</td>
<td>DASD</td>
</tr>
<tr>
<td>COPY2 ARCHIVE LOG DEVICE TYPE (UNIT2)</td>
<td>'BLANK'</td>
</tr>
<tr>
<td>SPACE ALLOCATION METHOD (ALCUNIT)</td>
<td>CYLINDER</td>
</tr>
<tr>
<td>PRIMARY SPACE ALLOCATION (PRIQTY)</td>
<td>100</td>
</tr>
<tr>
<td>SECONDARY SPACE ALLOCATION (SECQTY)</td>
<td>10</td>
</tr>
<tr>
<td>ARCHIVE LOG BLOCK SIZE IN BYTES (BLKSIZE)</td>
<td>24,576</td>
</tr>
<tr>
<td>MAXIMUM READ TAPE UNITS (MAXRTU)</td>
<td>2</td>
</tr>
<tr>
<td>TAPE UNIT DEALLOCATION PERIOD (DEALLCT)</td>
<td>0000:00</td>
</tr>
<tr>
<td>MAX NUMBER OF DATASETS RECORDED IN BSDS (MAXARCH)</td>
<td>1,000</td>
</tr>
<tr>
<td>FIRST ARCHIVE COPY MASS STG GROUP NAME</td>
<td>'NONE'</td>
</tr>
<tr>
<td>SECOND ARCHIVE COPY MASS STG GROUP NAME</td>
<td>'NONE'</td>
</tr>
<tr>
<td>DAYS TO RETAIN ARCHIVE LOG DATA SETS (ARCRETN)</td>
<td>30</td>
</tr>
<tr>
<td>ISSUE WTOR BEFORE MOUNT FOR ARCHIVE VOLUME (ARCWTOR)</td>
<td>YES</td>
</tr>
<tr>
<td>COMPACT DATA (COMPACT)</td>
<td>NO</td>
</tr>
<tr>
<td>QUIESCE PERIOD (QUIESCE)</td>
<td>5</td>
</tr>
<tr>
<td>SINGLE VOLUME (SVOLARC)</td>
<td>NO</td>
</tr>
</tbody>
</table>

CATALOG ARCHIVE DATASETS (CATALOG)

The alias of the VSAM integrated catalog facility user catalog or the name of the master catalog where the DB2 VSAM data sets created during installation are cataloged. The MVS catalog alias is also used as the high-level qualifier for DB2 VSAM data sets.

Install parameter CATALOG ALIAS on panel DSNTIPA, or ZPARM CATALOG in DSN6ARVP.

Field Name: QWP3CTLG

COPY1 ARCHIVE LOG DEVICE TYPE (UNIT)

The device type or unit name for storing archive log data sets.

The value can be any alphanumeric string. If you choose to archive to DASD, you can specify a generic device type with a limited volume range. DB2 requires that all archive log data sets allocated on DASD are cataloged.

If the device type is DASD, CATALOG DATA must be set to YES. If the unit name specifies DASD, the archive log data sets can extend to a maximum of 15 volumes. PRIQTY and SECQTY must be large enough to contain all active log data set data without extending beyond 15 volumes. If the unit name specifies a tape device, DB2 can extend to a maximum of 20 volumes. Default is TAPE.

Install parameter DEVICE TYPE 1 on panel DSNTIPA, or ZPARM UNIT in DSN6ARVP.
Archive Log Installation Parameters (DSNTIPA)

Field Name: QWP3UNT1

COPY2 ARCHIVE LOG DEVICE TYPE (UNIT2)

Indicates the device type or unit name for storing the second copy of archive log data sets.

The value can be any alphanumeric string. If you choose to archive to DASD, you can specify a generic device type with a limited volume range. DB2 requires that all archive log data sets allocated on DASD are cataloged.

If the device type is DASD, then CATALOG DATA must be set to YES. If the unit name specifies DASD, the archive log data sets can extend to a maximum of 15 volumes. PRIQTY and SECQTY must be large enough to contain all active log data set data without extending beyond 15 volumes. If the unit name specifies a tape device, DB2 can extend to a maximum of 20 volumes. Default is TAPE.

Install parameter DEVICE TYPE 2 on panel DSNTIPA, or ZPARM UNIT2 in DSN6ARVP.

Field Name: QWP3UNT2

SPACE ALLOCATION METHOD (ALCUNIT)

The unit used in allocating archive data sets. Possible values are:

- CYLINDER
  - Space allocation by cylinders (QWP3CYL=1)
- TRACKS
  - Space allocation by tracks (QWP3TRCK=1)
- BLOCKS
  - Space allocation by blocks (QWP3CYL=0 and QWP3TRCK=0)

Install parameter ALLOCATION UNITS on panel DSNTIPA, or ZPARM ALCUNIT in DSN6ARVP.

Field Name: RT0106SA

PRIMARY SPACE ALLOCATION (PRIQTY)

The primary space allocation for archive data sets.

Install parameter PRIMARY QUANTITY on installation panel DSNTIPA, or ZPARM PRIQTY in DSN6ARVP.

Field Name: QWP3RISP

SECONDARY SPACE ALLOCATION (SECQTY)

The amount of DASD secondary space allocation for an archive log data set.

The units used are specified by the ALLOCATION UNITS field. When blank (default), the CLIST calculates this space using block size and size of the log.

Install parameter SECONDARY QTY on panel DSNTIPA, or ZPARM SECQTY in DSN6ARVP.

Field Name: QWP3SECS

ARCHIVE LOG BLOCK SIZE IN BYTES (BLKSIZE)

The block size of the archive log data set.
The block size must be compatible with the device type used for archive logs. The value is rounded up to the next multiple of 4096 bytes.

If the archive log is written to tape, use the largest possible block size to improve the reading speed.

Recommended block size values are 28672 for tape, 20480 for 3380, and 24576 for 3390 or RAMAC.

Install parameter BLOCK SIZE on panel DSNTIPA, or ZPARM BLKSIZE in DSN6ARVP.

Field Name: QWP3BKSZ

MAXIMUM READ TAPE UNITS (MAXRTU)

The maximum number of tape units that can be allocated for archive read purposes.

Install parameter READ TAPE UNITS on panel DSNTIPA, or ZPARM MAXRTU in DSN6LOGP.

Field Name: QWP2MRTU

TAPE UNIT DEALLOCATION PERIOD (DEALLCT)

The number of minutes an archive read tape unit can remain unused before it is deallocated.

When archive log data is read from tape, this value should be high enough to allow DB2 to optimize tape handling for multiple read applications.

Install parameter DEALLOC PERIOD on panel DSNTIPA, or ZPARM DEALLCT in DSN6LOGP.

Field Name: QWP2DMIN

MAX NUMBER OF DATASETS RECORDED IN BSDS (MAXARCH)

The maximum number of archive log volumes that can be recorded in the BSDS.

When this number is exceeded, recording resumes at the beginning of the BSDS.

For dual archive, this value applies to each log data set. As an example, a value of 500 allows 500 COPY-1 and 500 COPY-2 data sets in the BSDS.

You must create image copies of all DB2 objects, probably several times, before the archive log data sets are discarded. If you fail to retain an adequate number of archive log data sets for all the image copies, you might need to cold start or reinstall DB2. In either case, data is lost.

Install parameter RECORDING MAX on panel DSNTIPA, or ZPARM MAXARCH in DSN6LOGP.

Field Name: QWP2ARCL

FIRST ARCHIVE COPY MASS STG GROUP NAME

The mass storage system volume group name of the first storage group.

Field Name: QWP3MSV1

SECOND ARCHIVE COPY MASS STG GROUP NAME

The mass storage system volume group name of the second storage group.

Field Name: QWP3MSV2
Archive Log Installation Parameters (DSNTIPA)

DAYS TO RETAIN ARCHIVE LOG DATA SETS (ARCRETN)

The number of days DB2 keeps archive log data sets.

This value is added to the current date to calculate the expiration date.

The retention period is often used in tape management systems to control the reuse and scratching of data sets and tapes. DB2 uses this as the value for the dynamic allocation parameter DALRETPD when archive log data sets are created.

Install parameter RETENTION PERIOD on panel DSNTIPA, or ZPARM ARCRETN in DSN6ARVP.

Field Name: QWP3RETN

ISSUE WTOR BEFORE MOUNT FOR ARCHIVE VOLUME (ARCWTOR)

Indicates whether DB2 must send a message to the operator and wait for an answer before attempting to mount an archive log data set.

Other DB2 users can be forced to wait while the mount is pending. They are not affected while DB2 is waiting for a response to the message.

When YES, a device such as tape is used that requires long delays for mounts. DEVICE TYPE 1 shows the device type or unit name.

Install parameter WRITE TO OPER on panel DSNTIPA, or ZPARM ARCWTOR in DSN6ARVP.

Field Name: QWP3WTOR

COMPACT DATA (COMPACT)

Indicates whether data written to archive logs is compacted.

This option only applies to data written to a 3480 device that has the improved data recording capability (IDRC) feature.

Install parameter COMPACT DATA on panel DSNTIPA, or ZPARM COMPACT in DSN6ARVP.

Field Name: QWP3COMP

QUIESCE PERIOD (QUIESCE)

The maximum amount of time (in seconds) permitted for DB2 to attempt a full system quiesce.

Install parameter QUIESCE PERIOD on panel DSNTIPA, or ZPARM QUIESCE in DSN6ARVP.

Field Name: QWP3MQP

SINGLE VOLUME (SVOLARC)

Indicates whether single-volume DASD archives are used.

Install parameter SINGLE VOLUME on panel DSNTIPA, or ZPARM SVOLARC in DSN6ARVP.

Field Name: QWP3SVOL
This topic shows detailed information about “System Parameters - Buffer Pool Parameters (DSNTIP1)”. This block shows the default buffer pools for user data and indexes.

**System Parameters - Buffer Pool Parameters (DSNTIP1)**

The field labels shown in the following sample layout of “System Parameters - Buffer Pool Parameters (DSNTIP1)” are described in the following section.

---

**DEFAULT 4-KB BUFFER POOL FOR USER DATA (TBSBPOOL)**

The name of the 4 KB buffer pool for user table spaces.

Install parameter DEFAULT BUFFER POOL FOR USER DATA on installation panel DSNTIP1, or ZPARM TBSBPOOL in DSN6SYSP.

**Field Name:** QWP1TBPL

**DEFAULT 8-KB BUFFER POOL FOR USER DATA (TBSBP8K)**

The default 8 KB buffer pool for:
- Table spaces with an 8 KB page size in implicitly created databases
- Explicitly created table spaces with an 8 KB page size, but without a buffer pool clause that is specified in the create table space statement.

Install parameter DEFAULT 8-KB BUFFER POOL FOR USER DATA on panel DSNTIP1 or ZPARM TBSBP8K in DSN6SYSP.

**Field Name:** QWP1TP8

**DEFAULT 16-KB BUFFER POOL FOR USER DATA (TBSBP16K)**

The default 16 KB buffer pool for:
- Table spaces with a 16 KB page size in implicitly created databases
- Explicitly created table spaces with a 16 KB page size, but without a buffer pool clause that is specified in the create table space statement.

Install parameter DEFAULT 16-KB BUFFER POOL FOR USER DATA on panel DSNTIP1 or ZPARM TBSBP16K in DSN6SYSP.

**Field Name:** QWP1TP16

**DEFAULT 32-KB BUFFER POOL FOR USER DATA (TBSBP32K)**

The default 32 KB buffer pool for:
- Table spaces with a 32 KB page size in implicitly created databases
- Explicitly created table spaces with a 32 KB page size, but without a buffer pool clause that is specified in the create table space statement.

Install parameter DEFAULT 32-KB BUFFER POOL FOR USER DATA on panel DSNTIP1 or ZPARM TBSBP32K in DSN6SYSP.

**Field Name:** QWP1TP32
Buffer Pool Parameters (DSNTIP1)

DEFAULT BUFFER POOL FOR USER INDEXES (IDXBPOOL)

The name of the 4 KB buffer pool used for indexes on user data.

Install parameter DEFAULT BUFFER POOL FOR USER INDEXES on installation panel DSNTIP1, or ZPARM IDXBPOOL in DSN6SYSP.

Field Name: QWP1IXPL
Data Definition Control Support (DSNTIPZ)

This topic shows detailed information about “System Parameters - Data Definition Control Support (DSNTIPZ)

This shows the installation and configuration for data definition control support.

Two SQL tables (application and object registration) are identified and created even if data definition control support is not installed. This simplifies future activation of the facility. Specified application identifiers (DB2 plans or collections of packages) can be registered in the application registration table and, optionally, their associated DB2 object names can be registered in the object registration table. DB2 consults these two tables prior to accepting a given DDL statement to make sure that a particular application identifier and object name are registered.

Fields in this block can contain long names. When a long name exceeds the available space, it is truncated, the parameter identifier and the full name are printed in a separate list at the end of the report.

System Parameters - Data Definition Control Support (DSNTIPZ)

The field labels shown in the following sample layout of “System Parameters - Data Definition Control Support (DSNTIPZ)” are described in the following section.

DATA DEFINITION CONTROL SUPPORT (DSNTIPZ)

INSTALL DD CONTROL (RGFINSTL)................................NO
CONTROL ALL APPLICATIONS (RGFDEDPL)..........................NO
REQUIRE FULL NAMES (RGFFULLQ)...............................YES
UNREGISTERED DDL DEFAULT (RGFDEFLT)......................ACCEPT
REGISTER TABLE OWNER (RGFCOLID)........................DSNZPARM
DDL REGISTRATION DATABASE NAME (RGFDBNAM)..............DSNRGFDB
APPL REGISTRATION TABLE NAME (RGFNMPT).................DSNZPARMRGFNMPRTD
OBJECT REGISTRATION TABLE NAME (RGFMORT).............DSNZPARMRGFNMORTD
ESCAPE CHARACTER (RGFESCP)................................X'40'

INSTALL DD CONTROL (RGFINSTL)

Indicates whether data definition support has been installed.

Install parameter INSTALL DD CONTROL SUPT on panel DSNTIPZ, or ZPARM RGFINSTL in DSN6SPRM.

Field Name: QWP4REGI

CONTROL ALL APPLICATIONS (RGFDEDPL)

Indicates that the DB2 system is completely controlled by a set of closed applications identified in the application registration table.

Closed applications require their DB2 objects to be managed solely through the plans or packages registered in the application registration table.

Install parameter CONTROL ALL APPLICATIONS on panel DSNTIPZ, or ZPARM RGFDEDPL in DSN6SPRM.

Field Name: QWP4REGD

REQUIRE FULL NAMES (RGFFULLQ)

Indicates whether registered objects require fully qualified names.
Data Definition Control Support (DSNTIPZ)

Install parameter REQUIRE FULL NAMES on panel DSNTIPZ, or ZPARM RGFFULLQ in DSN6SPRM.

Field Name: QWP4REGQ

UNREGISTERED DDL DEFAULT (RGFDEFLT)

The action taken for DDL that names an unregistered object.

Options are REJECT, ACCEPT, or APPL, which rejects the DDL when the current application is not registered.

Install parameter UNREGISTERED DDL DEFAULT on panel DSNTIPZ, or ZPARM RGFDEFLT in DSN6SPRM.

Field Name: QWP4REGU

REGISTER TABLE OWNER (RGFCOLID)

The owner of the application registration table and the object registration table.

This identifier can be a long string. If there is insufficient space to show the complete string, the string is truncated in the report block. The complete string is shown in a separate list of long names at the end of the report.

Install parameter REGISTRATION OWNER on panel DSNTIPZ, or ZPARM RGFCOLID in DSN6SPRM.

Field Name: QWP4REGC

DDL REGISTRATION DATABASE NAME (RGFDBNAM)

The name of the database that contains the registration tables.

Install parameter REGISTRATION DATABASE on panel DSNTIPZ, or ZPARM RGFDBNAM in DSN6SPRM.

Field Name: QWP4REGN

APPL REGISTRATION TABLE NAME (RGFNMPRT)

The name of the application registration table.

Install parameter APPL REGISTRATION TABLE on panel DSNTIPZ or ZPARM RGFNMPRT in DSN6SPRM.

Field Name: QWP4REGA

OBJECT REGISTRATION TABLE NAME (RGFMORT)

The name of the object registration table.

Install parameter OBJT REGISTRATION TABLE on panel DSNTIPZ, or ZPARM RGFNMORT in DSN6SPRM.

Field Name: QWP4REGO

ESCAPE CHARACTER (RGFESCP)

The escape character used in the application registration table (ART) or object registration table (ORT).

Sets of names in the ART and ORT can be represented by patterns that use the underscore (_) and percent sign (%) characters in the same way as in an SQL LIKE predicate.

Install parameter ART/ORT ESCAPE CHARACTER on panel DSNTIPZ, or ZPARM RGFESCP in DSN6SPRM.
Data Definition Control Support (DSNTIPZ)

Field Name: QWP4ESC
This topic shows detailed information about “System Parameters - Define Group or Member (DSNTIPK)”.  

This panel shows the members in a data sharing group.

DB2 subsystems that share data must belong to a DB2 data sharing group, which runs on a Parallel Sysplex. A data sharing group is a collection of one or more DB2 subsystems that access shared DB2 data. A Parallel Sysplex is a collection of MVS systems that communicate and cooperate with each other.

**System Parameters - Define Group or Member (DSNTIPK)**

The field labels shown in the following sample layout of “System Parameters - Define Group or Member (DSNTIPK)” are described in the following section.

`DEFINE GROUP OR MEMBER (DSNTIPK)`

---

- **GROUP NAME (GRPNNAME)**: The name of the DB2 data sharing group.
  - The group name encompasses the entire data sharing group and is the basis for the coupling facility structure names.
  - N/A means this DB2 is not part of a data sharing group.
  - Install parameter GROUP NAME on panel DSNTIPK, or ZPARM GRPNAME in DSN6GRP.
  - **Field Name**: QWPAGRN

- **MEMBER NAME (MEMBNAME)**: The member name of this DB2.
  - N/A means this DB2 is not part of a data sharing group.
  - Install parameter MEMBER NAME on panel DSNTIPK, or ZPARM MEMBNAME in DSN6GRP.
  - **Field Name**: QWPAMBRN

- **DEL CF STRUCTS (DEL_CFSTRUCTS_ON_RESTART)**: Shows whether to:
  - Delete change-data (CD) structures during restart
  - Attempt to delete coupling-facility (CF) structures, including shared communications area (SCA) structures, internal resource lock manager (IRLM lock) structures, and allocated group buffer pools.
  - This field corresponds to field DEL CF STRUCTS on installation panel DSNTIPK.
  - ZPARM name DEL_CFSTRUCTS_ON_RESTART in DSN6SYSP.
  - **Field Name**: QWP1DCFS

- **PARALLELISM ASSISTANT (ASSIST)**:
Define Group or Member (DSNTIPK)

Shows whether this DB2 member can assist a parallelism coordinator with parallel processing.

When YES, this member is considered an assistant at both bind and run time. To be a viable assistant at run time, both the VPPSEQT and VPXPSEQT buffer pool thresholds of this member must be greater than 0.

N/A means this DB2 is not part of a data sharing group.

Install parameter ASSISTANT on panel DSNTIPK or ZPARM ASSIST in DSN6GRP.

Field Name: QWPAASST

PARALLELISM COORDINATOR (COORDNTR)

Shows whether this DB2 member can coordinate parallel processing on other members of the group.

When NO, a query can be processed by this DB2 member only.

When YES, a read-only query running on this DB2 member can be processed in part on other members of the group.

N/A means this DB2 is not part of a data sharing group.

Install parameter COORDINATOR on panel DSNTIPK or ZPARM COORDNTR in DSN6GRP.

Field Name: QWPACOOR
Databases and Spaces Started Automatically (DSNTIPS)

This topic shows detailed information about “System Parameters - Databases and Spaces Started Automatically (DSNTIPS)”. This block shows the databases, table spaces, and index spaces that are started or restarted automatically when DB2 is started. ZPARM ALL/dbname in DSN6SPRM.

System Parameters - Databases and Spaces Started Automatically (DSNTIPS)

The field labels shown in the following sample layout of “System Parameters - Databases and Spaces Started Automatically (DSNTIPS)” are described in the following section.

DATABASES AND SPACES STARTED AUTOMATICALLY (DSNTIPS)

--

ALL

DATABASE NAME

The name of a database that is to be started automatically.

Field Name: QWP8DBNM

SPACE NAME

Contains the name of a table space or index space that is to be started automatically.

Field Name: QWP8SPNM
Default Startup Modules (DSNTIPO3)

This topic shows detailed information about “System Parameters - Default Startup Modules (DSNTIPO3)”.

System Parameters - Default Startup Modules (DSNTIPO3)

The field labels shown in the following sample layout of “System Parameters - Default Startup Modules (DSNTIPO3)” are described in the following section.

DEFAULT STARTUP MODULES (DSNTIPO3)

<table>
<thead>
<tr>
<th>PARAMETER MODULE</th>
<th>DSNZPARM</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACCESS CONTROL (ACCESS_CNTL_MODULE)</td>
<td>DSNX@XAC</td>
</tr>
<tr>
<td>IDENTIFY/AUTH (IDAUTH_MODULE)</td>
<td>DSN3@ATH</td>
</tr>
<tr>
<td>SIGNON (SIGNON_MODULE)</td>
<td>DSN3@SGN</td>
</tr>
</tbody>
</table>

PARAMETER MODULE

Shows the name of the active subsystem parameter module.

This field corresponds to field PARAMETER MODULE on installation panel DSNTIPO3.

Field Name: QWP1ZPNM

ACCESS CONTROL (ACCESS_CNTL_MODULE)

Shows the name of the default access control exit module.

This field corresponds to field ACCESS CONTROL on installation panel DSNTIPO3. The ZPARM name is ACCESS_CNTL_MODULE in DSN6SYSP.

Field Name: QWP1DXAC

IDENTIFY/AUTH (IDAUTH_MODULE)

Shows the name of the default identify or authorization exit module.

This field corresponds to field IDENTIFY/AUTH on installation panel DSNTIPO3. The ZPARM name is IDAUTH_MODULE in DSN6SYSP.

Field Name: QWP1DATH

SIGNON (SIGNON_MODULE)

Shows the name of the default signon exit module.

This field corresponds to field SIGNON on installation panel DSNTIPO3. The ZPARM name is SIGNON_MODULE in DSN6SYSP.

Field Name: QWP1DSGN
Distributed Data Facility Panel 1 (DSNTIPR)

This topic shows detailed information about “System Parameters - Distributed Data Facility Panel 1 (DSNTIPR)”.

This block shows how Distributed Data Facility (DDF) was started and the names used to connect another DB2 subsystem.

To use DDF, you must have VTAM installed, even if you use TCP/IP connections only.

System Parameters - Distributed Data Facility Panel 1 (DSNTIPR)

The field labels shown in the following sample layout of “System Parameters - Distributed Data Facility Panel 1 (DSNTIPR)” are described in the following section.

DISTRIBUTED DATA FACILITY PANEL 1 (DSNTIPR)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DDF STARTUP OPTION (DDF)</td>
<td>Indicates whether DDF is loaded, and if so, how it was started.</td>
</tr>
<tr>
<td>RLFERRD</td>
<td>Provides information about resource limit specification table access error.</td>
</tr>
<tr>
<td>RESYNCHRONIZATION INTERVAL IN MINUTES (RESYNC)</td>
<td>Specifies the synchronization interval for DDF.</td>
</tr>
<tr>
<td>CMTSTAT</td>
<td>Describes the status of the DBAT.</td>
</tr>
<tr>
<td>IDTHTOIN</td>
<td>Indicates the idle thread timeout interval.</td>
</tr>
<tr>
<td>EXTSEC</td>
<td>Specifies extended security options.</td>
</tr>
<tr>
<td>MAXTYPE1</td>
<td>Sets the maximum number of inactive type 1 threads.</td>
</tr>
</tbody>
</table>

**DDF STARTUP OPTION (DDF)**

Indicates whether DDF is loaded, and if so, how it was started.

- **AUTO** means DDF was loaded and started automatically when DB2 was started. The DDF address space was started as part of DDF initialization.
- **COMMAND** means DDF was initialized and the DDF address space was started at DB2 startup. If DDF is running, it was started from the console with the -DSN1 START DDF command. If it is not running, it can be started with this command.

Install parameter DDF STARTUP OPTION on panel DSNTIPR, or ZPARM DDF in DSN6FAC.

Field Name: QWP9STRT

**RLST ACCESS ERROR (RLFERRD)**

Shows what DB2 does when the governor cannot access the resource limit specification table or when no row in the table matches the currently executing statement.

- **NOLIMIT** (default) allows all dynamic SQL statements to run without limit.
- **NORUN** terminates all dynamic SQL statements immediately with an SQL error code.

A number from 1 to 5000000 is the default limit; if the limit is exceeded, the SQL statement is terminated.

Install parameter RLST ACCESS ERROR on panel DSNTIPR, or ZPARM RLFERRD in DSN6FAC.

Field Name: QWP9RLER
RESYNCHRONIZATION INTERVAL IN MINUTES (RESYNC)

The number of minutes between resynchronization periods.

A resynchronization period is the time during which indoubt logical units of work involving this DB2 subsystem and partner logical units are processed.

Install parameter RESYNC INTERVAL on panel DSNTIPR, or ZPARM RESYNC in DSN6FAC.

Field Name: QWP9RYC

DBAT STATUS (CMTSTAT)

Shows whether DB2 inactivates threads that have successfully committed or rolled back, and hold no cursors.

ACTIVE provides the best performance but consumes system resources.

INACTIVE is recommended when the installation must support a large number of connections.

When a thread becomes eligible for inactivation, DB2 tries to make it a type 2 inactive thread, which uses less storage than a type 1 inactive thread. If this fails, DB2 tries to make it a type 1 inactive thread. If neither attempt is successful, the thread remains active.

Install parameter DDF THREADS on panel DSNTIPR, or ZPARM CMTSTAT in DSN6FAC.

Field Name: QWP9CMST

IDLE THREAD TIMEOUT INTERVAL (IDTHTOIN)

The approximate time, in seconds, that an active server thread can remain idle before it is canceled.

Inactive and indoubt threads are not subject to timeout.

Threads are checked for timeouts every 3 minutes. This means that timeouts might not be honored for up to 3 minutes when the timeout value is less than this.

0 (default) means timeout processing is disabled, idle server threads remain in the system and continue to hold their resources, if any.

Install parameter IDLE THREAD TIMEOUT on panel DSNTIPR, or ZPARM IDTHTOIN in DSN6FAC.

Field Name: QWP9TTO

EXTENDED SECURITY (EXTSEC)

Extended security options.

When YES (strongly recommended), detailed reason codes are returned to a DRDA level 3 client when a DDF connection request fails because of security errors. When using SNA protocols, the requester must have included a product that supports the extended security sense codes, such as DB2 Connect version 5 and subsequent releases.

RACF users can change their passwords using the DRDA change password function. This support is only for DRDA level 3 requesters that have implemented support for changing passwords.
YES allows properly enabled DRDA clients to determine the cause of security failures without requiring DB2 operator support.

When NO, generic error codes are returned to the clients and RACF users are prevented from changing their passwords.

Install parameter EXTENDED SECURITY on panel DSNTIPR, or ZPARN EXTSEC in DSN6SYS.

Field Name: QWP1SCER

MAX TYPE 1 INACTIVE THREADS (MAXTYPE1)

Indicates the number of type 1 inactive threads that DB2 allows.

A large number of type 1 inactive threads can adversely affect system performance. Type 1 inactive threads are used for DB2 private protocol.

DRDA uses type 2 inactive threads.

Zero indicates that type 1 inactive connections are not allowed. Threads remain active when they become eligible to be made a type 1 inactive thread.

A value greater than zero indicates that type 1 inactive connections are allowed, but are limited to this number. When a thread becomes eligible to be made a type 1 inactive thread, and this threshold is reached, the remote connection is terminated.

When this is equal to MAX REMOTE CONNECTED on panel DSNTIPE, DB2 allows all remote threads to become type 1 inactive threads.

Install parameter MAX INACTIVE DBATS on panel DSNTIPR, or ZPARM MAXTYPE1 in DSN6FAC.

Field Name: QWP9MAX1
This topic shows detailed information about “System Parameters - Distributed Data Facility Panel 2 (DSNTIP5)”. This block shows how Distributed Data Facility (DDF) was started and the names used to connect to another DB2 subsystem.

To use DDF, you must have VTAM installed, even if you use TCP/IP connections only.

**Note**: The fields shown on this panel depend on the installed DB2 version.

**System Parameters - Distributed Data Facility Panel 2 (DSNTIP5)**

The field labels shown in the following sample layout of “System Parameters - Distributed Data Facility Panel 2 (DSNTIP5)” are described in the following section.

```
DISTRIBUTED DATA FACILITY PANEL 2 (DSNTIP5)
-------------------------------------------
TCP/IP ALREADY VERIFIED (TCPALVER)..................YES
EXTRA BLOCKS REQ (EXTRAREQ)..........................100
EXTRA BLOCKS SRV (EXTRASRV)..........................100
TCP/IP KEEPALIVE (TCPKPALV)..........................120
CONNECTION QUEUE MAX DEPTH (MAXCONQN)...............0
CONNECTION QUEUE MAX WAIT (MAXCONQW).................0
POOL THREAD TIMEOUT (POOLINAC)......................120
HOP SITE AUTHORIZATION (HOPAUTH)......................BOTH
```

**TCP/IP ALREADY VERIFIED (TCPALVER)**

Indicates whether DB2 accepts TCP/IP connection requests containing only a user ID.

When YES, a connection request is accepted with a user ID only. This value must be the same for all members of a data sharing group.

When NO (default), TCP/IP clients must provide authentication information (password, RACF PassTicket, or Kerberos ticket) to gain access to DB2.

Install parameter TCP/IP ALREADY VERIFIED on panel DSNTIP5, or ZPARM TCPALVER in DSN6FAC.

**Field Name**: QWP9TCPA

**EXTRA BLOCKS REQ (EXTRAREQ)**

The maximum number of extra DRDA query blocks DB2 requests from a remote DRDA server.

The default is 100.

This controls the total amount of data that can be transmitted on any given network exchange. It does not limit the size of the SQL query answer set.

Install parameter EXTRA BLOCKS REQ on panel DSNTIP5, ZPARM EXTRAREQ in DSN6SYSP.

**Field Name**: QWP1EXBR

**EXTRA BLOCKS SRV (EXTRASRV)**
Distributed Data Facility Panel 2 (DSNTIP5)

The maximum number of extra DRDA query blocks DB2 returns to a DRDA client.

The default is 100.

This controls the total amount of data that can be transmitted on any given network exchange. It does not limit the size of the SQL query answer set.

Install parameter EXTRA BLOCKS SRV on panel DSNTIP5, ZPARM EXTRASRV in DSN6SYSP.

Field Name: QWP1EXBS

TCP/IP KEEPALIVE (TCPKPALV)

Indicates whether the TCP/IP configuration KeepAlive value has been overwritten.

When ENABLE (default), KeepAlive is enabled, the TCP/IP configuration stack value is used.

When DISABLE, TCP/IP KeepAlive has been disabled.

A value in the range 1 through 65534 means KeepAlive is active, and the TCP/IP stack value has been overridden. The number reported shows the time, in seconds, between TCP/IP probes.

When considering overwriting the keep-alive time, it is recommended to set a value close to the IDLE THREAD TIMEOUT value on installation panel DSN7IPR or the IRLM RESOURCE TIMEOUT value on installation panel DSN7IPI. It is good practice to set all these to about five minutes, or less.

Because KeepAlive detection is accomplished by probing the network at this interval, avoid small values, which can cause excessive network traffic and system resource consumption.

The trick is to find a proper balance that allows network failures to be detected on a timely basis without impacting system and network performance.

Install parameter TCP/IP KEEPALIVE on panel DSNTIP5, ZPARM TCPKPALV in DSN6FAC.

Field Name: QWP9TCKA

CONNECTION QUEUE MAX DEPTH (MAXCONQN)

The maximum depth of the connection-request queue of connections that are waiting for a DBAT to process a request. If this value is non-zero, and QWP9CMST is active, or the subsystem is not a member of a data sharing group, DB2 operates as if this value were 0. This field corresponds to field CONN QUEUE MAX DEPTH on installation panel DSNTIP5. The ZPARM name is MACONQN in DSN6FAC.

Field Name: QWP9MCONQN

CONNECTION QUEUE MAX WAIT (MAXCONQW)

The maximum time that a connection waits for a DBAT request. If this value is non-zero, and QWP9CMST is active, or the subsystem is not a member of a data sharing group, DB2 operates as if this value is 0.

This field corresponds to field CONN QUEUE MAX WAIT on installation panel DSNTIP5. The ZPARM name is MAXCONQW in DSN6FAC.
POOL THREAD TIMEOUT (POOLINAC)

The approximate time, in seconds, that a DBAT can remain idle in the pool before it is terminated.

A DBAT thread in the pool counts as an active thread against MAX REMOTE ACTIVE and can hold locks, but does not have any cursors.

Threads are checked for timeouts every 3 minutes. This means that timeouts might not be honored for up to 3 minutes when the timeout value is less than this. The default is 120.

Install parameter POOL THREAD TIMEOUT on panel DSNTIP5, ZPARM POOLINAC in DSN6FAC.

HOP SITE AUTHORIZATION (HOPAUTH)

Indicates whose authorization is checked at a second server (sometimes called a hop site) when the requester is not DB2 for z/OS.

This option applies only when DB2 private protocol is used for the hop from the second to the third site.

When BOTH (default), the package owner's authorization is checked for static SQL, and the runner's authorization ID is checked for dynamic SQL.

When RUNNER, both static and dynamic SQL use the runner's authorization.

Install parameter AUTH AT HOP SITE on panel DSNTIP5, ZPARM HOPAUTH in DSN6SPRM.
DB2 Utilities Parameters (DSNTIP6, DSNTIP61, DSNTIP62)

This topic shows detailed information about “System Parameters - DB2 Utilities Parameters (DSNTIP6, DSNTIP61, DSNTIP62)”.

This block shows the default behavior of enhancements to the BACKUP SYSTEM and other utilities.

System Parameters - DB2 Utilities Parameters (DSNTIP6, DSNTIP61, DSNTIP62)

The field labels shown in the following sample layout of “System Parameters - DB2 Utilities Parameters (DSNTIP6, DSNTIP61, DSNTIP62)” are described in the following section.

DB2 UTILITIES PARAMETERS (DSNTIP6,DSNTIP61,DSNTIP62)
----------------------------------------------------
TEMPORARY UNIT NAME (VOLTDEVT)............................SYSDA
STATISTICS HISTORY (STATHIST)..............................NONE
STATISTICS ROLLUP (STATROLL)................................YES
UTILITY TIMEOUT FACTOR (UTIMOUT)..............................6
TEMPLATE TIME (TEMPLATE_TIME).................................UTC
MAXIMUM DEGREE OF UTILITY PARALLELISM (PARAMDEG_UTIL)....99
SYSTEM-LEVEL BACKUPS (SYSTEM_LEVEL_BACKUPS)..................NO
RESTORE/RECOVER (RESTORE_RECOVER_FROMDUMP)...................NO
DUMP CLASS NAME (UTILS_DUMP_CLASS_NAME)....................'BLANK'
MAXIMUM TAPE UNITS (RESTORE_TAPEUNITS).......................0
DEFAULT TEMPLATE (FCCOPYDDN)...................................
DBE1.&DB..&SN..N&DSNUM..&U
REORG PART SORT NPSI (REORG_PART_SORT_NPSI)...................AUTO
REORG LIST PROCESSING (REORG_LIST_PROCESSING)...............PARALLEL
REORG MAPPING DATABASE (REORG_MAPPING_DATABASE)............N/P
REORG DROP PBG PARTS (REORG_DROP_PBG_PARTS)..................NO
REORG IGNORE FREE SPACE (REORG_IGNORE_FREESPACE)............NO
UTILITY CACHE OPTION (SEQPRES).............................YES
STATISTICS CLUSTERING (STATCLUS).........................ENHANCED

TEMPORARY UNIT NAME (VOLTDEVT)

Shows the device type or unit name for allocating temporary data sets. It is the direct access or disk unit name used for the precompiler, compiler, assembler, sort, linkage editor, and utility work-files in the tailored jobs and CLISTS.

It can be any device type acceptable to the DYNALLOC parameter of the SORT or OPTION options for DFSORT.

The default is SYSDA.

Install parameter TEMPORARY UNIT NAME on DSNTIPA2, or ZPARM VOLTDEVT in DSN6SPRM.

Field Name: QWP4VDTY

STATISTICS HISTORY (STATHIST)

Shows which inserts and updates are recorded in catalog history tables.

The report can show the following values:

N / NONE
Changes in the catalog are not recorded. This is the default.

A / ALL
All inserts and updates in the catalog are recorded.
DB2 Utilities Parameters (DSNTIP6, DSNTIP61, DSNTIP62)

P / ACCESSPATH
All inserts and updates to access path related catalog statistics are recorded.

S / SPACE
All inserts and updates to space related catalog statistics are recorded.

Install parameter STATISTICS HISTORY on panel DSNTIPO, or ZPARM STATHIST in DSN6SPRM.

Field Name: QWP4STHT

STATISTICS ROLLUP (STATROLL)
Shows whether RUNSTATS utility aggregates the partition level statistics, even though some parts may not contain data.
This should be YES for DB2 systems that have large partitioned table spaces, index spaces, or both. This enables the aggregation of part level statistics and helps the optimizer to choose a better access path.
Install parameter STATISTICS ROLLUP on panel DSNTIPO, or ZPARM STATROLL in DSN6SPRM.

Field Name: QWP4STRL

UTILITY TIMEOUT FACTOR (UTIMOUT)
Shows how much longer utilities can wait for a resource than SQL applications can.
This is the number of RESOURCE TIMEOUT units that a utility or utility command can wait for a lock or for all claims on a resource of a particular claim class to be released. The default value is 6, meaning a utility can wait 6 times longer than an SQL application for a resource.
Install parameter UTILITY TIMEOUT on panel DSNTIPI, or ZPARM UTIMOUT in DSN6SPRM.

Field Name: QWP4UTO

TEMPLATE TIME (TEMPLATE_TIME)
Specifies the default setting for the TIME option of the template utility control statement. Possible values are:
- UTC (utility control)
- Local
This field corresponds to field TEMPLATE TIME on installation panel DSNTIP6. The ZPARM name is TEMPLATE_TIME in DSN6SPRM.

Field Name: QWP4TPTM

MAXIMUM DEGREE OF UTILITY PARALLELISM (PARAMDEG_UTIL)
The maximum degree of utility parallelism.

Field Name: QWP4UMD

SYSTEM-LEVEL BACKUPS (SYSTEM_LEVEL_BACKUPS)
Shows if RECOVER uses system level backups as the recovery base.
Install parameter SYSTEM-LEVEL BACKUPS on installation panel DSNTIP6, or ZPARM SYSTEM_LEVEL_BACKUPS in DSN6SPRM.
Field Name: QWP4SLBU

RESTORE/RECOVER (RESTORE_RECOVER_FROMDUMP)

If YES, the system-level backup that is the recovery base, is from a dump on tape. Otherwise NO is shown.

Install parameter RESTORE/RECOVER on installation panel DSNTIP6, or ZPARM RESTORE_RECOVER_FROMDUMP in DSN6SPRM.

Field Name: QWP4RRFD

DUMP CLASS NAME (UTILS_DUMP_CLASS_NAME)

The name of the DFSMSHSM dump class used by the restore system utility to restore from a system-level backup that has been dumped to tape.

Install parameter DUMP CLASS NAME on installation panel DSNTIP6, or ZPARM UTILS_DUMP_CLASS_NAME in DSN6SPRM.

Field Name: QWP4RSDC

MAXIMUM TAPE UNITS (RESTORE_TAPEUNITS)

The maximum number of tape units or tape drives that the restore system utility can use to restore from a system-level backup that has been dumped to tape.

Install parameter MAXIMUM TAPE UNITS on installation panel DSNTIP6, or ZPARM RESTORE_TAPEUNITS in DSN6SPRM.

Field Name: QWP4RSMT

DEFAULT TEMPLATE (FCCOPYDDN)

Specifies the default setting of the FCCOPYDDN subsystem parameter for the COPY, LOAD, REBUILD INDEX, REORG INDEX, and REORG TABLESPACE utility control statements when the FLASHCOPY parameter is YES or CONSISTENT. FCCOPYDDN specifies a DB2 utility template data-set name expression that is used to derive the copy data-set name that is allocated by the utility during operation.

This field corresponds to field DEFAULT TEMPLATE on installation panel DSNTIP6. The ZPARM name is FCOPYDDN in DSN6SPRM.

Field Name: QWP4FCCD

REORG PART SORT NPSI (REORG_PART_SORT_NPSI)

Specifies the default method of building a non-partitioned secondary index during the REORG tablespace part. This setting is used when the SORTNPSI keyword is not specified in a utility control statement.

Possible values are:

- Auto
- Disable
- Enable

This field corresponds to field REORG PART SORT NPSI in installation panel DSNTIP61. The ZPARM name is REORG_PART_SORT_NPSI in DSN6SPRM.

Field Name: QWP4RPSN

REORG LIST PROCESSING (REORG_LIST_PROCESSING)
DB2 Utilities Parameters (DSNTIP6, DSNTIP61, DSNTIP62)

Specifies the default value for the REORG TABLESPACE PARALLEL option.

- Parallel
- Serial

The ZPARM name is REORG_LIST_PROCESSING in DSN6SPRM.

Field Name: QWP4RLPR

REORG MAPPING DATABASE (REORG_MAPPING_DATABASE)

The default database in which REORG TABLESPACE SHRLEVEL change implicitly creates the mapping table. This field corresponds to field RECORD MAPPING DB on installation panel DSNTIP61. The ZPARM name RECORD_MAPPING_TABLE in DSN6SPRM.

Field Name: QWP4RMDB

REORG DROP PBG PARTS (REORG_DROP_PBG_PARTS)

If YES, REORG completes, REORG drops empty, and trailing partitions are set in a PARTITION-BY-GROWTH table space.

This field corresponds to field REORG DROP PBG PARTS on INSTALLATION panel DSNTIP61. The ZPARM name is REORG_DROP_PBG_PARTS in DSN6SPRM.

Field Name: QWP4RPBG

REORG IGNORE FREE SPACE (REORG_IGNORE_FREESPACE)

YES indicates that REORG tablespace does not use the PCTFREE and FREEPAGE values when it reloads data rows into a partition-by-growth (PBG) table space if:

- A subset of the partitions is reorganized.
- The associated table contains LOB columns that cause a REORG AUX NO REQUEST to fail.

This field corresponds to field REORG IGNORE FREESPACE in installation panel DSNTIP61. ZPARM name is REORG_IGNORE_FREESPACE in DSN6SPRM.

Field Name: QWP4RIFS

UTILITY CACHE OPTION (SEQPRES)

Shows whether utilities that scan a nonpartitioned index followed by an update of a subset of the pages in the index allow data to remain in 3990 cache longer when reading data.

Install parameter UTILITY CACHE OPTION on panel DSNTIPE, or ZPARM SEQPRES in DSN6PRM.

Field Name: QWP4PST

STATISTICS CLUSTERING (STATCLUS)

Shows if the RUNSTATS utility uses enhanced or standard clustering statistics: ENHANCED is used if it is on, otherwise STANDARD is shown.

Install parameter STATISTICS CLUSTERING on panel DSNTIP6, or ZPARM STATCLUS in DSN6SPRM.

Field Name: QWP4STCL
This topic shows detailed information about “System Parameters - DB2 Version Install (DSNTIPA1)”.

**System Parameters - DB2 Version Install (DSNTIPA1)**

The field labels shown in the following sample layout of “System Parameters - DB2 Version Install (DSNTIPA1)” are described in the following section.

**DB2 VERSION INSTALL (DSNTIPA1)**

```
DB2 VERSION INSTALL (DSNTIPA1)
------------------------------
DATA SHARING ENABLED (DSHARE) ..............................NO
INSTALL TYPE (NEWFUN) .....................................YES
```

**DATA SHARING ENABLED (DSHARE)**

Indicates whether data sharing is enabled.

Install parameter DATA SHARING on panel DSNTIPA1, or ZPARM DSHARE in DSN6GRP.

**Field Name:** QWPAIOPT

**INSTALL TYPE (NEWFUN)**

If YES, the DB2 subsystem/group is running in New Function Mode. At this mode/catalog level, the New Function Mode is enabled and available. The DB2 catalog is completely Unicode (UTF-8) and long names can be used.

Install parameter INSTALL TYPE on panel DSNTIPA1, or ZPARM NEWFUN in DSNHDECP.

**Field Name:** QWPBNEWF
This topic shows detailed information about “System Parameters - IRLM Installation Parameters (DSNTIPI)”.

This block shows the installation of the internal resource lock manager (IRLM). There is one IRLM for each DB2 subsystem.

**Note:** The fields shown on this panel depend on the installed DB2 version.

**System Parameters - IRLM Installation Parameters (DSNTIPI)**

The field labels shown in the following sample layout of “System Parameters - IRLM Installation Parameters (DSNTIPI)” are described in the following section.

**IRLM INSTALLATION PARAMETERS (DSNTIPI)**

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IRLM SUBSYSTEM NAME (IRLMSID)</td>
<td>The IRLM subsystem name defined to MVS. This is used for communication between DB2 and the IRLM. It is included in the MVS subsystem table IEFSSN xx, where xx is the value of SUBSYSTEM MEMBER on installation panel DSNTIPM. If the IRLM for IMS is installed, the DB2 IRLM name is different because two IRLMs on the same MVS system must have unique names. Install parameter SUBSYSTEM NAME on panel DSNTIPI, or ZPARM IRLMSID in DSN6SPRM.</td>
</tr>
<tr>
<td>IRLM RESOURCE TIMEOUT IN SECONDS (IRLMRWT)</td>
<td>The number of seconds before a timeout is detected. This is an integer multiple of DEADLOCK TIME on panel DSNTIPJ. Timeout means that a lock request has waited for a resource (or for claims on a resource for a particular claim class to be released) longer than this time. For data sharing, the actual timeout period is longer than the timeout value. Install parameter RESOURSE TIMEOUT on panel DSNTIPI, or ZPARM IRLMRWT in DSN6SPRM.</td>
</tr>
<tr>
<td>IRLM AUTOMATIC START (IRLMAUT)</td>
<td>Indicates whether IRLM is started automatically by DB2.</td>
</tr>
</tbody>
</table>
IRLM Installation Parameters (DSNTIPI)

Install parameter AUTO START on panel DSNTIPI, or ZPARM IRLMAUT in DSN6SPRM.

Field Name: QWP4IAUT

IRLM START PROCEDURE NAME (IRLMPRC)

The name of the IRLM procedure invoked by MVS if AUTO START is YES.
The name cannot be the same as the subsystem name given for
SUBSYSTEM NAME.
Install parameter PROC NAME on panel DSNTIPI, or ZPARM IRLMPRC in DSN6SPRM.

Field Name: QWP4IPRC

SECONDS DB2 WILL WAIT FOR IRLM START (IRLMSWT)

The IRLM wait time in seconds.
DB2 autostart abends if IRLM does not start within this time.
Install parameter TIME TO AUTOSTART on panel DSNTIPI, or ZPARM IRLMSWT in DSN6SPRM.

Field Name: QWP4ISWT

U LOCK FOR REPEATABLE READ OR READ STABILITY (RRULOCK)

Indicates whether the U (UPDATE) lock is used when using repeatable read (RR) or read stability (RS) isolation to access a table.
When YES, the U lock is used for an updated cursor with repeatable read or read stability.
When NO, the S lock is used for an updated cursor with repeatable read or read stability. If the cursor in the running applications includes the clause FOR UPDATE OF, but updates are infrequent, S locks generally provide better performance.
Install parameter U LOCK FOR RR/RS on panel DSNTIPI, or ZPARM RRULOCK in DSN6SPRM.

Field Name: QWP4RRU

X LOCK FOR SEARCHED UPDATE/DELETE (XLKUPDLT)

The locking method used when performing a searched UPDATE or DELETE.
When NO, DB2 uses an S or U lock when scanning for qualifying rows. For any qualifying rows or pages the lock is upgraded to an X lock before performing the update or delete. For nonqualifying rows or pages the lock is released if using ISOLATION(CS). For ISOLATION(RS), or ISOLATION(RR), an S lock is retained on the rows or pages until the next commit point. This option is used to achieve higher rates of concurrency.
When YES, DB2 gets an X lock on qualifying rows or pages. For ISOLATION(CS), the lock is released if the rows or pages are not updated or deleted. For ISOLATION(RS) or ISOLATION(RR), an X lock is retained until the next commit point. This is beneficial in a data sharing environment when most or all searched updates and deletes use an index. The downside is that if searched updates or deletes result in a tablespace scan, the likelihood of timeouts and deadlocks greatly increases.
IRLM Installation Parameters (DSNTIPI)

Install parameter X LOCK FOR SEARCHED U/D on panel DSNTIPI, or ZPARM XLKUPDLT in DSN6SPRM.

**Field Name:** QWP4XLUD

**IMS/BMP TIMEOUT FACTOR (BMPTOUT)**

The number of RESOURCE TIMEOUT units that an IMS BMP connection waits for a lock to be released.

The default value is 4, meaning that an IMS BMP connection can wait 4 times the resource timeout value for a resource.

Install parameter IMS BMP TIMEOUT on panel DSNTIPI, or ZPARM BMPTOUT in DSN6SPRM.

**Field Name:** QWP4WBMP

**IMS/DLI TIMEOUT FACTOR (DLITOUT)**

The number of RESOURCE TIMEOUT units that a DL/I batch connection waits for a lock to be released.

The default value is 6, meaning that an IMS BMP connection can wait 4 times the resource timeout value for a resource.

Install parameter DL/I BATCH TIMEOUT on panel DSNTIPI, or ZPARM DLITOUT in DSN6SPRM.

**Field Name:** QWP4WDLI

**WAIT FOR RETAINED LOCKS (RETLWAIT)**

Indicates whether a request is suspended until an incompatible retained lock becomes available.

This value is only significant in a data sharing environment. It indicates how long a transaction should wait for a lock on a resource if another DB2 in the data sharing group has failed and is holding an incompatible lock on that resource. Locks held by failed DB2 members are called retained locks.

This value is a multiplier that is applied to the connection's normal timeout value. For example, if the retained lock multiplier is 2, then the timeout period for a call attachment connection that is waiting for a retained lock is twice the normal CAF timeout period. The default is 0, meaning applications do not wait for incompatible retained locks, the lock request is immediately rejected and the application receives a "resource unavailable" SQLCODE.

Install parameter RETAINED LOCK TIMEOUT on panel DSNTIPI, or ZPARM RETLWAIT in DSN6SPRM.

**Field Name:** QWP4WAIT

**ENABLE DB CHECKING**

Enable database checking.

**Field Name:** QWP4DBCK
IRLM Processing Parameters

This topic shows detailed information about “System Parameters - IRLM Processing Parameters”.

This block shows the system parameters for internal resource lock manager (IRLM) processing.

Note: The fields shown on this panel depend on the installed DB2 version.

System Parameters - IRLM Processing Parameters

The field labels shown in the following sample layout of “System Parameters - IRLM Processing Parameters” are described in the following section.

IRLM PROCESSING PARAMETERS

----------------------------------------
WAIT TIME FOR LOCAL DEADLOCK........5,000
NUMBER OF LOCAL CYCLES PER GLOBAL CYCLE........1
TIMEOUT INTERVAL..........................30
IRLM MAXIMUM CSA USAGE ALLOWED........0
Z/OS LOCK TABLE HASH ENTRIES........1,048,576
PENDING NUMBER OF HASH ENTRIES........0
Z/OS LOCK TABLE LIST ENTRIES........8,282
MAX 31-BIT IRLM PRIVATE STORAGE........0
MAX 64-BIT IRLM PRIVATE STORAGE........0

WAIT TIME FOR LOCAL DEADLOCK

Wait time for local deadlock.

Field Name: QWP5DLOK

NUMBER OF LOCAL CYCLES PER GLOBAL CYCLE

Number of local cycles per global cycle.

Field Name: QWP5DCYC

TIMEOUT INTERVAL

Timeout interval.

Field Name: QWP5TVAL

IRLM MAXIMUM CSA USAGE ALLOWED

The maximum amount of common service area that can be used by IRLM.

The amount of space needed for the common service area (CSA) below the 16 MB line is less than 40 KB for each DB2 subsystem and 24 KB for each IRLM. High concurrent activity, parallelism, or high contention can require more CSA.

Most of the DB2 common data resides in the extended common service area (ECSA). Most modules, control blocks, and buffers reside in the extended private area. A DB2 subsystem with 200 concurrent users and 2000 open data sets should need less than 2 MB of virtual storage below the 16 MB line.

Field Name: QWP5MCSA

Z/OS LOCK TABLE HASH ENTRIES

The number of z/OS lock table hash entries.

Field Name: QWP5HASH
IRLM Processing Parameters

PENDING NUMBER OF HASH ENTRIES
The number of z/OS lock table hash entries pending.
Field Name: QWP5PHSH

Z/OS LOCK TABLE LIST ENTRIES
The number of z/OS lock table list entries.
Field Name: QWP5RLE

MAX 31-BIT IRLM PRIVATE STORAGE
The maximum amount of 31-bit IRLM private storage that is available of
the 2 GB virtual storage limit, for normal operations in IRLM. IRLM
reserves an additional 10% of the 2 GB for use by requests in IRLM.
Field Name: QWP5BPM

MAX 64-BIT IRLM PRIVATE STORAGE
The maximum amount of 64-bit IRLM private storage that is available of
the total amount of storage that is specified by MEMLIMIT, for normal
operations in IRLM. IRLM reserves an additional 10% of the amount that is
specified by MEMLIMIT for use by requests in IRLM.
Field Name: QWP5APM
This topic shows detailed information about “System Parameters - Lock Escalation Parameters (DSNTIPJ)”.

This panel shows the characteristics of IRLM time-sharing fields and other locking options.

The default values are adequate for most sites in normal conditions.

**System Parameters - Lock Escalation Parameters (DSNTIPJ)**

The field labels shown in the following sample layout of “System Parameters - Lock Escalation Parameters (DSNTIPJ)” are described in the following section.

<table>
<thead>
<tr>
<th>LOCK ESCALATION PARAMETERS (DSNTIPJ)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAX PAGE OR ROW LOCKS PER TABLE SPACE (NUMLKTS) ...........1,000</td>
</tr>
<tr>
<td>MAX PAGE OR ROW LOCKS PER USER (NUMLKUS) .................10,000</td>
</tr>
</tbody>
</table>

**MAX PAGE OR ROW LOCKS PER TABLE SPACE (NUMLKTS)**

The default (SYSTEM) for the LOCKMAX clause of the SQL statements CREATE TABLESPACE and ALTER TABLESPACE.

Install parameter LOCKS PER TABLE(SPACE) on panel DSNTIPJ, or ZPARM NUMLKTS in DSN6SPRM.

**Field Name:** QWP4LKTS

**MAX PAGE OR ROW LOCKS PER USER (NUMLKUS)**

The maximum number of page or row locks that a single application can hold concurrently on all table spaces.

This includes locks on data pages, index pages, subpages, and rows that the program acquires when it accesses table spaces.

The limit applies to all table spaces defined with the LOCKSIZE PAGE, LOCKSIZE ROW, or LOCKSIZE ANY options. 0 means that there is no limit to the number of page and row locks a program can acquire.

DB2 assumes that 250 bytes of storage are required for each lock. If NO is specified for CROSS MEMORY, the value of this field has to take into account the available lock space. If referential constraints between tables is defined, the value of this field might need to be increased.

Install parameter LOCKS PER USER on panel DSNTIPJ, or ZPARM NUMLKUS in DSN6SPRM.

**Field Name:** QWP4LKUS
Log Installation Parameters (DSNTIPL, DSNTIPH)

This topic shows detailed information about “System Parameters - Log Installation Parameters (DSNTIPL, DSNTIPH)”.

This block shows the characteristics of active log data sets.

System Parameters - Log Installation Parameters (DSNTIPL, DSNTIPH)

The field labels shown in the following sample layout of “System Parameters - Log Installation Parameters (DSNTIPL, DSNTIPH)” are described in the following section.

LOG INSTALLATION PARAMETERS (DSNTIPL,DSNTIPH)
----------------------------------------------------------
OUTPUT BUFFER SIZE IN K BYTES (OUTBUFF) ................. 4,000
DBM1 STORAGE FOR FAST LOG (LOGAPSTG) .................... 100
CHECKPOINT FREQUENCY (CHKFREQ) .......................... 500,000
UR CHECK FREQUENCY (URCHKTH) ..................... 0
UR LOG RECORD WRITTEN THRESHOLD IN KB (URLGWTH) ........ 0
LIMIT BACKOUT (LBACKOUT) ................................ AUTO
BACKOUT DURATION (BACKODUR) ............................ 5
PSEUDO-CLOSE FREQUENCY (PCLOSEN) ......................... 5
PSEUDO-CLOSE TIMER (PCLOSET) .............................. 10
CHECKPOINTS BETWEEN LEVEL ID UPDATES (DLDFTQP) .... 5
NUMBER OF ACTIVE LOG COPIES (TWOACTV) ..................... 1
NUMBER OF ARCHIVE LOG COPIES (TWOARCH) .............. 1
COPY 1 PREFIX (ARCPFX1) .................................. DSN911.ARCHLOG1
COPY 2 PREFIX (ARCPFX2) .................................. DSN911.ARCHLOG2
TIMESTAMP ARCHIVE LOG DATA SETS (TSTAMP) ............. YES
CHECKPOINT TYPE (CHKTYPE) ................................. SINGLE
RECORDS/CHECKPOINT (CHKLOGR) .......................... N/P
MINUTES/CHECKPOINT (CHKMINS) .......................... N/P

OUTPUT BUFFER SIZE IN K BYTES (OUTBUFF)

The output log buffer size in kilobytes.

There is only one output log buffer per DB2 subsystem.

Increasing this parameter reduces BSDS I/O updates when there is a buffer wraparound. Frequent wraparounds are likely in LOAD or REORG with logging, and mass insert operations.

Increasing this parameter also helps avoid log write waits for an available buffer during heavy update workload.

When the specified size is not a 4 KB multiple, it is rounded up to the next 4 KB multiple.

Install parameter OUTPUT BUFFER on DSNTIPL, or ZPARM OUTBUFF in DSN6LOGP.

Field Name: QWP2OBPS

DBM1 STORAGE FOR FAST LOG (LOGAPSTG)

The maximum DBM1 storage that can be used by the fast log apply process. The default value is 0 MB, which means that fast log apply is disabled except during DB2 restart, when fast log apply is always enabled.

Install parameter LOG APPLY STORAGE on panel DSNTIPL, or ZPARM LOGAPSTG in DSN6SYSYP.

Field Name: QWP1FLBZ
Log Installation Parameters (DSNTIPL, DSNTIPH)

CHECKPOINT FREQUENCY (CHKFREQ)

Checkpoint frequency. This shows either the number of minutes (1 through 60) or the number of DB2 log records between the start of successive checkpoints. DB2 starts a new checkpoint when this value is reached.

You can use the SET LOG command to change the number of log records between checkpoints dynamically. Valid values are 1-60 when specifying a time value and 200-16000000 when specifying a number of records.

Install parameter CHECKPOINT FREQ on panel DSNTIPL, ZPARM CHKFREQ in DSN6SYSP.

Field Name: QWP1LOGL

UR CHECK FREQUENCY (URCHKTH)

Shows the number of checkpoint cycles to complete before DB2 issues a warning message to the console and writes an IFCID 313 record for an uncommitted, indoubt, or inflight unit of recovery (UR). The default is 0, which disables this option.

Install parameter UR CHECK FREQ on panel DSNTIPL, or ZPARM URCHKTH in DSN6SYSP.

Field Name: QWP1URCK

UR LOG RECORD WRITTEN THRESHOLD IN KB (URLGWTH)

Shows the number of log records that are to be written by an uncommitted unit of recovery (UR) before DB2 issues a warning message to the console. This provides notification of a long-running UR. Long-running URs might result in a lengthy DB2 restart or a lengthy recovery situation for critical tables. Log records are specified in 1-K (1000 log records) increments. A value of 0 indicates that no write check is to be performed.

Install parameter UR LOG WRITE CHECK on panel DSNTIPL, ZPARM URLGWTH in DSN6SYSP.

Field Name: QWP1LWCK

LIMIT BACKOUT (LBACKOUT)

Shows whether some backward log processing should be postponed.

When NO, DB2 backward log processing processes all inflight units of recovery (URs) and URs for abending transactions.

When YES, DB2 postpones backout processing for some units of work until the command RECOVER POSTPONED is issued.

AUTO (default) postpones some backout processing but automatically starts the backout processing when DB2 restarts and begins accepting new work.

When YES or AUTO, backout processing runs concurrently with new work. Page sets or partitions with backout work pending are unavailable until their backout work is complete.

Install parameter LIMIT BACKOUT on panel DSNTIPL, or ZPARM LBACKOUT in DSN6SYSP.

Field Name: QWP1LMBO

BACKOUT DURATION (BACKODUR)
Indicates how much of the log to process for backout when LIMIT BACKOUT = YES or AUTO.

During restart, backward log processing continues until both of the following events occur:
- All inflight and inabort URs with update activity against the catalog or directory are backed out.
- The number of log records processed is equal to the number specified in BACKOUT DURATION times the value of CHECKPOINT FREQ. If the checkpoint frequency is specified in minutes, the number of records processed is the default of 50000 records multiplied by the value of CHECKPOINT FREQ.

In-flight and in-abort URs that are not completely backed out during restart are converted to postponed-abort status. Page sets or partitions with postponed-backout work are put into restart pending (RESTP). This state blocks all access to the object other than access by the command RECOVER POSTPONED or by automatic backout processing performed by DB2 when LIMITED BACKOUT = AUTO.

A table space might be in restart pending mode, without the associated index spaces also in restart pending mode. This happens if a postponed abort UR makes updates only to non-indexed fields of a table in a table space. In this case, the indexes are accessible to SQL (for index-only queries), even though the table space is inaccessible.

Install parameter BACKOUT DURATION on panel DSNTIPL, or ZPARM BACKODUR in DSN6SYSP.

Field Name: QWP1BDUR

**PSEUDO-CLOSE FREQUENCY (PCLOSEN)**

The number of consecutive DB2 checkpoints that a page set or partition can remain in read/write mode since it was last updated. When this limit or the RO SWITCH TIME is reached, DB2 changes the page set or partition to read only.

This can improve performance for recovery, logging, and data-sharing processing.

Install parameter RO SWITCH CHKPTS on panel DSNTIPL, or ZPARM PCLOSEN in DSN6SYSP.

Field Name: QWP1FREQ

**PSEUDO-CLOSE TIMER (PCLOSET)**

The number of minutes that a page set or partition can remain in read-write mode since it was last updated. When this limit or the RO SWITCH CHKPTS is reached, DB2 changes the page set or partition to read-only.

This can improve performance for recovery, logging, and data-sharing processing.

Install parameter RO SWITCH TIME on panel DSNTIPL, or ZPARM PCLOSET in DSN6SYSP.

Field Name: QWP1TMR

**CHECKPOINTS BETWEEN LEVEL ID UPDATES (DLDFREQ)**
Log Installation Parameters (DSNTIPL, DSNTIPH)

How often, in checkpoints, the level ID of a page set or partition is updated. When zero (0), downlevel detection is disabled.

Use the following criteria to decide on a suitable value for this parameter:

- **How often are backup and restore methods used outside of the DB2 control (such as DSN1COPY or DFDSS dump and restore)?** If rarely used, there is no need to update the level ID frequently.
- **How many page sets are open for update at the same time?** If DB2 updates level IDs frequently, there is extra protection against downlevel page sets. However, a performance degradation can occur if the level IDs for many page sets must be set at every checkpoint.
- **How often does the subsystem take checkpoints?** If the DB2 subsystem takes frequent system checkpoints, set the level ID frequency to a higher value.

Install parameter LEVELID UPDATE FREQ on panel DSNTIPL, or ZPARM DLDFREQ in DSN6SYSP.

**Field Name:** QWP1DFRQ

**NUMBER OF ACTIVE LOG COPIES (TWOACTV)**

The number of copies of the active log being maintained: 2 indicates dual logging.

**Field Name:** QWP2DUAL

**NUMBER OF ARCHIVE LOG COPIES (TWOARCH)**

The number of copies of the archive log being produced during offloading: 2 indicates dual logging.

Install parameter NUMBER OF COPIES on PANEL DSNTIPH, or ZPARM TWOARCH in DSN6LOGP.

**Field Name:** QWP2ADL

**COPY 1 PREFIX (ARCPFX1)**

The prefix of the first archive log data set.

Install parameter Archive Logs: COPY1 PREFIX on panel DSNTIPH, or ZPARM ARCPFX1 in DSN6ARVP.

**Field Name:** QWP3RE1N

**COPY 2 PREFIX (ARCPFX2)**

The prefix of the second archive log data set. If single logging is used, this value is a default.

Install parameter Archive Logs: COPY2 PREFIX on panel DSNTIPH, or ZPARM ARCPFX2 in DSN6ARVP.

**Field Name:** QWP3RE2N

**TIMESTAMP ARCHIVE LOG DATA SETS (TSTAMP)**

Indicates whether the date and time of creation of the DB2 archive log data set is included in the archive log data set name.

Possible values are:

- **YES (QWP3DTIM=1)**

  The maximum allowable length of the user-controlled portion of the archive log prefix is reduced from 35 characters to 19
Log Installation Parameters (DSNTIPL, DSNTIPH)

...characters. This allows the 16-character timestamp to be added to the archive log data set prefix. The timestamp format is as follows: DyydddThhhmmst, where:

- D Starts the date.
- yy Is the last two digits of the year.
- add Is the day of the year.
- T Starts the time.
- hh Is the hour.
- mm Are the minutes.
- ss Are the seconds.
- t Is the tenths of a second.

The maximum allowable length of the user-controlled portion of the archive log prefix is reduced from 35 characters to 19 characters. This reduction in size permits the 16-character date and time qualifiers (timestamp) to be added to the archive log data set prefix.

NO (QWP3DTIM=0 and QWP3DTFM=0)

The archive data set name does not contain a timestamp.

EXT (QWP3DTFM=1)

The archive data set name contains a timestamp with an extended date component in the format: .Dyyyyddd. A value of EXT in this field causes the lengths of the values that are entered for field COPY 1 PREFIX and field COPY 2 PREFIX to be audited to ensure that neither exceeds 17 bytes (19 bytes for other settings of TIMESTAMP ARCHIVES).

Install parameter TIMESTAMP ARCHIVES on panel DSNTIPH, or ZPARM TSTAMP in DSN6ARVP.

Field Name: RT0106AL

CHECKPOINT TYPE (CHKTYPE)

Shows the LOG checkpoint type. It can have the following values:

SINGLE

Either records or minutes.

BOTH

Both records and minutes, as specified by Records Between Checkpoint (QWP1LOGR) and Mins Between Checkpoint (QWP1LOGM).

ZPARM CHKTYPE in DSN6SYSP.

Field Name: QWP1LOGT

RECORDS/CHECKPOINT (CHKLOGR)

Shows the number of records between log checkpoints if the LOG checkpoint type is BOTH (records and minutes).

This field corresponds to field RECORDS/CHECKPOINT on installation panel DSNTIPL1, or ZPARM name CHKLOGR in DSN6SYSP.

Field Name: QWP1LOGR
Log Installation Parameters (DSNTIPL, DSNTIPH)

MINUTES/CHECKPOINT (CHKMINS)

Shows the number of minutes between log checkpoints if the LOG
checkpoint type is BOTH (records and minutes).

This field corresponds to field MINUTES/CHECKPOINT on installation
panel DSNTIPL1, or ZPARM name CHKMINS in DSN6SYSP.

Field Name: QWP1LOGM
List of Long Names

This topic shows detailed information about “System Parameters - List of Long Names”.

This block is printed at the end of the system parameters report when the report contains long names that have been truncated. The block shows the parameter identifier, in alphabetic order, and the complete name. If the name is too long to show on one line, it continues on the next.

System Parameters - List of Long Names

The field labels shown in the following sample layout of “System Parameters - List of Long Names” are described in the following section.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEFLTID</td>
<td>DSNZPARMDEFLTIDDSNZPARMDEFLTIDDSNZPARMDEFLTID</td>
</tr>
<tr>
<td>RGFCOLID</td>
<td>DSNZPARMRGFCOLIDDSNZPARMRGFCOLIDDSNZPARMRGFCOLID</td>
</tr>
<tr>
<td>RGFNMORT</td>
<td>DSNZPARMRGFNMORTDSNZPARMRGFNMORTDSNZPARMRGFNMORT</td>
</tr>
<tr>
<td>RGFNMPRT</td>
<td>DSNZPARMRGFNMPRTDSNZPARMRGFNMPRTDSNZPARMRGFNMPRT</td>
</tr>
<tr>
<td>FLFAUTH</td>
<td>DSNZPARMFLFAUTHDSNZPARMFLFAUTHDSNZPARMFLFAUTH</td>
</tr>
<tr>
<td>SYSADM2</td>
<td>DSNZPARMSYSADM2DSNZPARMSYSADM2DSNZPARMSYSADM2</td>
</tr>
<tr>
<td>SYSOPR1</td>
<td>DSNZPARMSYSOPR1DSNZPARMSYSOPR1DSNZPARMSYSOPR1DSNZPARMSYSOPR1</td>
</tr>
<tr>
<td>NZENDE</td>
<td>DSNZPARMSYSOPR1DSNZPARMSYSOPR1DSNZPARMSYSOPR1DSNZPARMSYSOPR1</td>
</tr>
<tr>
<td>SYSOPR2</td>
<td>ABCDEFGHIJKLMNOPQRSTUVWXYZ</td>
</tr>
</tbody>
</table>

Chapter 54. System Parameters Report Blocks 54-49
MVS Parmlib Update Parameters (DSNTIPM)

This topic shows detailed information about “System Parameters - MVS Parmlib Update Parameters (DSNTIPM)”.

This block shows the parameters used to produce the DSNTIJMV job that defined DB2 to MVS and updated the following PARMLIB members:

- **IEFSSN xx**
  - to define DB2 and IRLM as formal MVS subsystems

- **IEAAPF xx**
  - to authorize the prefix.SDSNLOAD, prefix.SDSNLINK, and prefix.SDSNEXIT libraries

- **LNKLST xx**
  - to include the prefix.SDSNLINK library.

### System Parameters - MVS Parmlib Update Parameters (DSNTIPM)

The field labels shown in the following sample layout of “System Parameters - MVS Parmlib Update Parameters (DSNTIPM)” are described in the following section.

```
MVS PARMLIB UPDATE PARAMETERS (DSNTIPM)
---------------------------------------
SUBSYSTEM DEFAULT (SSID) .................................D851
SUPPRESS SOFT ERRORS (SUPERRS)...............................NO

SUBSYSTEM DEFAULT (SSID)

The MVS subsystem name for DB2. The name is used in member IEFSSN xx of SYS1.PARMLIB.

A valid name has 1-4 characters, the first must be A-Z, #, $, or @. Others must be A-Z, 1-9, #, $, or @. Default is DSN1.

Install parameter SUBSYSTEM NAME on panel DSNTIPM, or ZPARM SSID in DSNHDECP.

Field Name: QWPBSSID

SUPPRESS SOFT ERRORS (SUPERRS)

Shows whether the recording of errors, such as invalid decimal data and arithmetic exceptions, in the operating system data set SYS1.LOGREC is suppressed.

When YES, these exceptions are not recorded in the LOGREC data set.

Install parameter SUPPRESS SOFT ERRORS on panel DSNTIPM or ZPARM SUPERRS in DSN6SPRM.

Field Name: QWP4SAE
Operator Functions Installation Parameters (DSNTIPO)

This topic shows detailed information about “System Parameters - Operator Functions Installation Parameters (DSNTIPO)”. This block shows various operator functions, such as write-to-operator route codes, automatic recall, and the maximum amount of CPU time allocated for a dynamic SQL statement.

Note: The fields shown on this panel depend on the installed DB2 version.

System Parameters - Operator Functions Installation Parameters (DSNTIPO)

The field labels shown in the following sample layout of “System Parameters - Operator Functions Installation Parameters (DSNTIPO)” are described in the following section.

OPERATOR FUNCTIONS INSTALLATION PARAMETERS (DSNTIPO)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>WTO ROUTE CODES (ROUTCDE)</td>
<td>The MVS console routing codes. These codes are assigned to messages that are not solicited from a specific console. Up to 16 comma-separated codes can be shown. Install parameter WTO ROUTE CODES on panel DSNTIPO, or ZPARM ROUTCDE in DSN6SYS. Field Name: QWP1SMRC</td>
</tr>
<tr>
<td>RESOURCE LIMIT FACILITY AUTOMATIC START (RLF)</td>
<td>Shows whether the resource limit facility (governor) is automatically started when DB2 is started. Install parameter RLF AUTO START on panel DSNTIPO, or ZPARM RLF in DSN6SYS. Field Name: QWP1RLF</td>
</tr>
<tr>
<td>RESOURCE LIMIT SPECIFICATION TABLE SUFFIX (RLFTBL)</td>
<td>The default resource limit specification table (RLST) suffix. This suffix is used when the resource limit facility (governor) is automatically started or when the governor is started without specifying a suffix. Install parameter RLST NAME SUFFIX on panel DSNTIPO, or ZPARM RLFTBL in DSN6SYS.</td>
</tr>
</tbody>
</table>
Operator Functions Installation Parameters (DSNTIPO)

Field Name: QWP1RLFT

RESOURCE LIMIT SPEC TABLE ERROR ACTION (RLFERR)

The action taken by DB2 when the governor cannot use the resource limit:

NOLIMIT

The dynamic SQL statements run without limit.

NORUN

The dynamic SQL statements terminated with an SQL error code.

A number from 1 to 5000000 represents the number of CPU service units allowed for a query.

Install parameter RLST ACCESS ERROR on panel DSNTIPO, or ZPARM RLFERR in DSN6SYSP.

Field Name: RLFERR

AUTO BIND (ABIND)

Indicates whether autobind is enabled. Values are:

YES

Allows automatic rebind operations to be performed when a plan/package:

• Was marked “invalid”.
• Was bound on DB2 Vn, but is now running on DB2 Vn-1
• After use on DB2 Vn-1 (as previously described), is later used again on DB2 Vn

NO

Prevent DB2 from performing any automatic rebind operations under any circumstances.

COEXIST

Allows automatic rebind operation to be performed in a DB2 Data Sharing coexistence environment when the plan/package:

• Is marked “invalid” or
• Was last bound in DB2 Vn and is running on DB2 Vn-1

Install parameter AUTO BIND on panel DSNTIPO, or ZPARM ABIND in DSN6SPRM.

Field Name: QWP4ABN

ALLOW EXPLAIN AT AUTOBIND (ABEXP)

Indicates whether EXPLAIN processing occurs during automatic rebind.

YES means EXPLAIN processing happens during automatic rebind of a plan or package that has EXPLAIN(YES) as a bind option. If the PLAN_TABLE does not exist, automatic rebind continues, but there is no EXPLAIN output. Explain processing does not happen for a plan or package with EXPLAIN(NO).

Install parameter EXPLAIN PROCESSING on panel DSNTIPO, or ZPARM ABEXP in DSN6SPRM.

Field Name: QWP4ABX

DPROP SUPPORT (EDPROP)

Shows whether DataPropagator NonRelational (DPROP) is used to propagate SQL changes made to tables defined with DATA CAPTURE CHANGES.
Operator Functions Installation Parameters (DSNTIPO)

1. No changes are propagated.

2. DPROP propagates SQL changes, and those changes made to tables defined with DATA CAPTURE CHANGES are only allowed when monitor trace class 6 is active, DPROP is installed, and the DB2 application is running in an IMS environment. If any of these conditions are not met, no changes to the DB2 table are permitted.

3. Data propagation occurs when monitor trace class 6 is active, DPROP is installed, and the DB2 application is running in an IMS environment. In this instance, an application that is not running in an IMS environment can update DB2 tables defined with DATA CAPTURE CHANGES. However, these changes are not propagated to IMS.

ANY  Allows subsystems to propagate some data with DPROP and other data with a different propagation program.

Tables that should only be updated by DB2 applications running in an IMS environment can be protected using the following methods:

- Use the ENABLE parameter on BIND to specify a specific attachment facility through which updates to data propagation tables can be made.
- Define a validation procedure for data propagation tables to define which plans can update those tables.
- Allow update authority for data propagation tables to a group of authorization IDs that can only run in IMS.

Install parameter DPROP SUPPORT on panel DSNTIPO, or ZPARM EDPROP and CHGDC in DSN6SPRM.

Field Name: QWP4ENF

SITE TYPE (SITETYP)

Shows whether this system is at a local site or a recovery site.

LOCALSITE

This is the site of the current system. Multiple image copies are made and are operational here. This is the default.

RECOVERYSITE

This an alternative site for recovery purposes.

The RECOVER utility uses this parameter to determine what site the current system is on and recovers everything from the copies of data registered at that site.

The RECOVER and MERGECOPY utilities use this to determine whether COPYDDN or RECOVERDDN is allowed with NEWCOPY NO.

Install parameter SITE TYPE on panel DSNTIPO, or ZPARM SITETYP in DSN6SPRM.

Field Name: QWP4MSTY

TRACKER SITE (TRKRSITE)

Indicates whether this subsystem is a remote tracker site for another DB2 subsystem.

When YES, this is a tracker site.
A DB2 tracker site is a separate DB2 subsystem or data sharing group that exists solely for the purpose of keeping shadow copies of your primary site's data. No independent work can be run on the tracker site.

Install parameter TRACKER TYPE on panel DSNTIPO, or ZPARM TRKRSITE in DSN6SPRM.

Field Name: QWP4TRKR

**READ COPY2 ARCHIVE (ARC2FRST)**

This field indicates whether the COPY2 archives should be read first when the DB2 subsystem is started.

Install parameter READ COPY2 ARCHIVE on DSNTIPO, or ZPARM ARC2FRST in DSN6LOGP.

Field Name: QWP2OPT2

**REAL TIME STATS (STATSINT)**

The time interval that DB2 waits before it attempts to write out page set statistics to the real-time statistics tables. This value is between 1 and 65535 minutes.

Install parameter REAL TIME STATS on panel DSNTIPO, or ZPARM STATSINT in DSN6SPRM.

Field Name: QWP4INTE

**STATISTICS FEEDBACK (STATFDBK_SCOPE)**

Specifies the scope of SQL statements for which DB2 is to recommend statistics. Possible values are:

- All
- Dynamic
- None
- Static

This value corresponds to field STATISTICS FEEDBACK on installation panel DSNTIPO. The ZPARM name is STATFDBK_SCOPE in DSN6SPRM.

Field Name: QWP4SFBS
This topic shows detailed information about “System Parameters - Other System Parameters”.

This block shows values not shown on other DB2 installation panels. These values are either set internally by DB2, or calculated from other install parameter values.

Note: The fields shown on this panel depend on the installed DB2 version.

System Parameters - Other System Parameters

The field labels shown in the following sample layout of “System Parameters - Other System Parameters” are described in the following section.

OTHER SYSTEM PARAMETERS

DUAL BSDS MODE (TWOBSDS)....................................YES
ROLL UP PARALLEL TASK ACCOUNTING (PTASKROL)............YES
NO. PAGES SMALL TABLE THRESHOLD (NPHTHRSH).....................0
OFFLOAD OPTION (OFFLOAD)......................................YES
SU CONVERSION FACTOR........................................281
MINIMUM DIVIDE SCALE (MINDVSCSCL)..............................NONE
STAR JOIN THRESHOLD (STJOINTH)................................10
ONLINE SYSTEM PARM USER ID MONITOR..........................N/P
ONLINE SYSTEM PARM CORREL ID MONITOR........................N/P
ONLINE SYSTEM PARM TIME CHANGED................................N/P
ONLINE SYSTEM PARM TYPE........................................N/P
DB2-SUPPLIED DECP INDICATOR......................................X’D5’
MAX CONCURRENT PKG OPS (MAX_CONCURRENT_PKG_OPS).............10
ADMIN SCHEDULER JCL PROC NAME (ADMTPROC)....................N/P
FREE LOCAL CACHED STATEMENTS (CACHEDYN_FREELOCAL)...........1
INDEX I/O PARALLELISM (INDEX_OPARALLELISM)..................YES
ZOSMETRICSD..................................................YES
USE TRACKMOD FOR IMPLICIT TS (IMPTKMOD).....................YES
DSSIZE FOR IMPLICIT TS (IMPSIZE).................................4
ENABLE MULTIPLE INDEX ACCESS (SUBQ_MIDX).......................YES
SPT01 INLINE LENGTH (SPT01_INLINE_LENGTH).....................32,138
DDF COMPATIBILITY (DDF_COMPATIBILITY)..........................NO
DYN STMT CACHE STOR (CACHE_DEP_TRACK_STOR_LIM)...............0
SMS DATACLASS NAME FOR TS (SMSDCFL)............................N/P
SMS DATACLASS NAME FOR IS (SMSDCIX)............................N/P
OUTER JOIN PERFORMANCE ENHANCEMENTS (OJPERFEH).............YES

DUAL BSDS MODE (TWOBSDS)

Shows whether two BSDS data sets are used.

A second BSDS (strongly recommended) makes recovery much easier in most situations. In cases that normally require recovery and restart, a second BSDS allows you to continue working. The storage overhead required is small and the data set is relatively inactive.

DB2 parameter TWOBSDS in DSN6LOGP.

Field Name: QWP2DBSD

ROLL UP PARALLEL TASK ACCOUNTING (PTASKROL)

Indicates whether DB2 generates a trace record at the originating task level that summarizes accounting information for all parallel tasks.

DB2 parameter PTASKROL in DSN6SYSP.

Field Name: QWP1PROL
Other System Parameters

NO. PAGES SMALL TABLE THRESHOLD (NPGTHRSH)

This parameter allows you to specify the optimizer threshold for qualifying a table as small.
-1 Every table qualifies as small.
0 No table qualifies as small (this is the default).
1 Only tables with zero pages qualify as small.
2 Tables with less than two pages qualify as small.
10 Tables with less than ten pages qualify as small.
502 Tables with less than 502 pages, and tables that have not had statistics collected qualify as small. For example, when NPages = -1.

DB2 parameter NPGTHRSH in DSN6SPRM.

Background and Tuning Information

Tables can be populated using insert just prior to their use by queries and then cleared immediately on completion of the queries. These tables are permanent even though the data they contain is transient.

This can cause problems when RUNSTATS is run overnight, or at other times when these tables are empty. This gives the optimizer the false indication that these tables contain no data when in fact, the tables will contain data when the query executes. This causes the optimizer to pick an inefficient access path. Usually the optimizer chooses to do a table scan, which would be the most efficient access path if the table were truly empty. Because the table is not empty when the query executes, it would be more efficient to use matching index access.

With this parameter, you can force the optimizer to treat tables containing no data as small tables. For these tables, the optimizer will:
- Select a matching index access rather than a table space scan and non-matching index access.
- Select the index with the most matching columns when more than one index qualifies for matching index access.
- Select indexes with the same number of matching columns on cost.

Field Name: QWP4NPAG

OFFLOAD OPTION (OFFLOAD)

Shows whether the offload process is initiated online.
ZPARM OFFLOAD in macro DSN6LOGP.
Field Name: QWP2OFFL

SU CONVERSION FACTOR

The CPU service unit conversion factor for this CPU.
This factor allows conversion CPU time in seconds to a common unit, called service unit (SU). The conversion factor used depends on the machine. Service units allow you to calculate CPU execution times across a data sharing group.

The conversion factor is used as follows:
Other System Parameters

CP secs * 16,000,000 / Conversion Factor = SUs
SUs * Conversion Factor / 16,000,000 = CP secs
This field does not map to an installation panel.

Field Name: QWPASUCV
MINIMUM DIVIDE SCALE (MINDVSCL)
The minimum scale for the result of a decimal division. The values for this parameter are none (the default), 3, or 6. If 3 or 6 is specified, this parameter overrides the DECDIV3 parameter.

Field Name: QWP4MDSC
STAR JOIN THRESHOLD (SJTABLES)
The minimum number of tables in the star schema query block, including the fact table, dimensions tables, and snowflake tables. This value is considered only if the subsystem parameter STARJOIN qualifies the query for star join.

Possible values are:
0     Star join is disabled. This is the default.
1, 2, or 3     Star join is always considered.
4 through 255     Star join is considered if the query block has at least the specified number of tables.
256 and greater     Star join is never considered.

DB2 parameter SJTABLES in DSN6SPRM.

Background and Tuning Information
Although star join can reduce bind time significantly it does not provide optimal performance in all cases. Performance of star join depends on a number of factors such as the available indexes on the fact table, the cluster ratio of the indexes, and the selectivity of rows through local and join predicates. Follow these general guidelines for setting the value of SJTABLES:

- If you have star schema queries with less than 10 tables and you want to make the star join method applicable to all qualified queries, set the value of SJTABLES to a low number, such as 5.
- If you have some star schema queries that are not necessarily suitable for star join but want to use star join for relatively large queries, use the default. The star join method will be considered for all qualified queries that have 10 or more tables.
- If you have star schema queries but normally do not want to use star join, you could increase SJTABLES, say to 15. This will greatly cut the bind time for large queries and avoid a potential bind time SQL return code -101 for large qualified queries.

Field Name: QWP4SJTB
ONLINE SYSTEM PARM USER ID MONITOR

The user ID that made the last online change to DB2 system settings.
Other System Parameters

Field Name: QWP4OZUS
ONLINE SYSTEM PARM CORREL ID MONITOR
The correlation ID of the online application that made the last change to DB2 system settings.
Field Name: QWP4OZCI
ONLINE SYSTEM PARM TIME CHANGED
Time of the last online change made to DB2 system settings.
Field Name: QWP4OZTM
ONLINE SYSTEM PARM TYPE
The type of DB2 system parameter changed by the last SET SYSPARM statement.
Field Name: QWP4OZTP
DB2-SUPPLIED DECP INDICATOR
Indicates that DECP is supplied by DB2.
Using a DB2 supplied DECP could cause data corruption due to applications using wrong CCSIDs.
Field Name: QWPBDB2S
MAX CONCURRENT PKG OPS (MAX_CONCURRENT_PKG_OPS)
The maximum number of package requests that can be processed simultaneously.
DB2 parameter MAX_CONCURRENT_PKG_OPS in DSN6SPRM.
Field Name: QWP4MXAB
ADMIN SCHEDULER JCL PROC NAME (ADMTPROC)
The name of the JCL procedure for starting the DB2 administrative scheduler task address space.
DB2 parameter ADMTPROC in DSN6SPRM.
Field Name: QWP4ADMT
FREE LOCAL CACHED STATEMENTS (CACHEDYN_FREELOCAL)
Indicates whether DB2 can free statements from the local dynamic statement cache to relieve storage constraints below the 2 GB bar. This parameter applies only for packages or plans that are bound with KEEPDYNAMIC(YES). Possible values are:

0  DB2 cannot free statements from the local cache
1  DB2 can free statements from the local cache

DB2 parameter CACHEDYN_FREELOCAL in DSN6SPRM.
Field Name: QWP4FRLC
INDEX I/O PARALLELISM (INDEX_IO_PARALLELISM)
The enablement of the index I/O parallelism ZPARM.
Field Name: QWP4IIOP
ZOSMETRICS
YES indicates that gathering of z/OS metrics using the RMF interface is enabled. ZPARM ZOSMETRICS in DSN6SPRM.

**Field Name:** QWP4METE

**USE TRACKMOD FOR IMPLICIT TS (IMPTKMOD)**

Shows whether you have specified the TRACKMOD option on ALTER TABLESPACE for an implicitly created table space.

This field corresponds to field TRACK MODIFIED PAGES on installation panel DSNTIP7. The ZPARM name is IMPTKMOD in DSN6SPRM.

**Field Name:** QWP1TKMD

**DSSIZE FOR IMPLICIT TS (IMPDSSIZE)**

Shows the maximum DSSIZE in gigabytes that DB2 uses for creating each partition of an implicitly created base table space.

This field corresponds to field DEFAULT DSSIZE on installation panel DSNTIP7. The ZPARM name is IMPDSSIZE in DSN6SPRM.

**Field Name:** QWP1DSSZ

**ENABLE MULTIPLE INDEX ACCESS (SUBQ_MIDX)**

Specifies whether to enable or disable multiple index access for queries that have subquery predicates:

- **NO** Disables multiple index access for queries.
- **YES** Enables multiple index access for queries.

The ZPARM name is SUBQ_MIDX IN DSN6SPRM.

**Field Name:** QWP4SQMX

**SPT01 INLINE LENGTH (SPT01_INLINE_LENGTH)**

The maximum length in bytes of LOB columns in the SPT01 directory space that are maintained in the base table. This field corresponds to field SPT01 INLINE LENGTH on installation panel DSNTIPA2. The ZPARM name is SPT01_INLINE_LENGTH in DSN6SPRM.

**Field Name:** QWP4S1IL

**DDF COMPATIBILITY (DDF_COMPATIBILITY)**

YES indicates that pre-DB2 10 behavior is used to determine the SQL types of stored procedure parameters in calls from non-Java clients. The ZPARM name is DDF_COMPATIBILITY and the ZPARM value is SP_PARM_NJV in DSN6FAC.

**Field Name:** QWP9SPPM

**DYN STMT CACHE STOR (CACHE_DEP_TRACK_STOR_LIM)**

Specifies the number of gigabytes of storage that DB2 allocates for hashing entries in the dynamic statement cache. This parameter can avoid storage shortages for long-running threads. The storage is allocated above the bar.

The ZPARM name is CACHE_DEP_TRACK_STOR_LIM in DSN6SPRM.

**Field Name:** QWP4CDTSL

**SMS DATACLASS NAME FOR TS (SMSDCFL)**
**Other System Parameters**

SMS data class for data table spaces. The data class name is a string of one to eight characters. The default is an empty string, which means that the SMS cluster is defined without the DATACLASS parameter.

When a valid data class name is specified, the SMS cluster is specified with the DATACLASS parameter using the name specified. If the name is not valid, SMS returns an error.

DB2 parameter SMSDCFL in DSN6SPRM.

**Field Name:** QWP4DCFS

**SMS DATACLASS NAME FOR IS (SMSDCIX)**

SMS data class for index table spaces. The data class name is a string of one to eight characters. The default is an empty string, which means that the SMS cluster is defined without the DATACLASS parameter.

When a valid data class name is specified, the SMS cluster is specified with the DATACLASS parameter using the name specified. If the name is not valid, SMS returns an error.

DB2 parameter SMSDCIX in DSN6SPRM.

**Field Name:** QWP4DCIX

**OUTER JOIN PERFORMANCE ENHANCEMENTS (OJPERFEH)**

Indicates whether outer join performance enhancements are enabled.

DB2 parameter OJPERFEH in DSN6SPRM.

**Field Name:** QWP4OJEH
Performance and Optimization (DSNTIP8, DSNTIP81)

This topic shows detailed information about “System Parameters - Performance and Optimization (DSNTIP8, DSNTIP81)”.

Note: The fields shown on this panel depend on the installed DB2 version.

System Parameters - Performance and Optimization (DSNTIP8, DSNTIP81)

The field labels shown in the following sample layout of “System Parameters - Performance and Optimization (DSNTIP8, DSNTIP81)” are described in the following section.

PERFORMANCE AND OPTIMIZATION (DSNTIP8,DSNTIP81)

-----------------------------------------------
CACHE DYNAMIC SQL (CACHEDYN)................................YES
OPTIMIZATION HINTS ALLOWED (OPTHINTS)........................NO
EVALUATE UNCOMMITTED (EVALUNC)..............................NO
SKIP UNCOMM INSERTS (SKIPUNCI)...............................NO
IMMEDWRITE OVERRIDE FLAG (IMMEDWRI)........................NO
REBIND PLANMGMT DEFAULT (PLANMGMT)..........................EXTENDED
PLANMGMTSCOPE DEFAULT (PLANMGMTSCOPE)........................STATIC
PACKAGE RELEASE COMMIT (PKGREL_COMMIT)........................YES
RANDOMIZE XML DOCID (XML_RANDOMIZE_DOCID)..................NO
BLOCK OPT 1 ROW SORT (OPT1ROWBLOCKSORT)........................NO
CURRENT DEGREE (CSSRDEF)......................................1
MAX DEGREE OF PARALLELISM (PARAMDEG)...........................0
MAX DEGREE FOR DPSI (PARAMDEG_DPSI).............................0
STAR JOIN ENABLING (STARJOIN)..................................DISABLE
MAX DATA CACHING IN MB (MXDTCACH).............................20
CURRENT REFRESH AGE (REFSHAGE)................................0
CURRENT MAINT TYPE (MAINTYPE).................................SYSTEM
UPDATE PART KEY COLS (PARTKEYU).................................YES
EDM BEST FIT (EDMBFIT)..........................................NO
VARCHAR FROM INDEX (RETVLCFK).................................NO

CACHE DYNAMIC SQL (CACHEDYN)

Indicates whether prepared dynamic SQL statements are saved for later use by eligible application processes in the EDM pool.

Install parameter CACHE DYNAMIC SQL on panel DSNTIP8, or ZPARM CACHEDYN in DSN6SPRM.

Field Name: QWP4CDYN

OPTIMIZATION HINTS ALLOWED (OPTHINTS)

Shows whether DB2 can use optimization hints from the PLAN_TABLE to influence the access paths used for certain queries.

Install parameter OPTIMIZATION HINTS on panel DSNTIP8, or ZPARM OPTHINTS in DSN6SPRM.

Field Name: QWP4HINT

EVALUATE UNCOMMITTED (EVALUNC)

Shows whether stage 1 predicate evaluation during table access can proceed upon uncommitted data or not.

This applies to isolation levels of Read Stability and Cursor Stability only.
When NO (default), predicate evaluation occurs only on committed data (or on the application's own uncommitted changes). NO ensures that all qualifying data is always included in the answer set.

When YES, predicate evaluation can occur upon uncommitted data. Only committed data is returned to the query. However, a decision can be made to omit a row from the answer set based on uncommitted data. Later, undo processing (statement rollback or statement failure) could cause the data to revert to a state that satisfies the predicate.

When YES, DB2 can request fewer locks than in previous versions when processing isolation level Read Stability and Cursor Stability queries. The number of locks avoided is related to the access path of the query, the number of rows evaluated when processing the stage 1 predicate of the query, and the number of those rows that are overflow rows. Specifically, for isolation level Read Stability and Cursor Stability queries, locks are avoided for rows that do not satisfy the stage 1 predicate, provided they are not overflow rows. Table access includes table space scans and index-to-data access, including ridlist-to-data access. For isolation Cursor Stability ridlist production, all row/page locking is avoided.

Install parameter EVALUATE UNCOMMITTED on panel DSNTIP8, or ZPARM EVALUNC in DSN6SPRM.

**Field Name:** QWP4EVUN

**SKIP UNCOMM INSERTS (SKIPUNCI)**

YES indicates that uncommitted inserts are treated as if they have not yet been executed. The ZPARM name is SKIPUNCI.

**Field Name:** QWP4SKUI

**IMMEDWRITE OVERRIDE FLAG (IMMEDWRI)**

Indicates how DB2 updates group buffer pool dependent pages. This is only valid in a data-sharing environment.

Group buffer pool dependent pages can be written to DASD or SYSTEM pagesets.

Values shown are:

- **NO** DB2 uses normal write activity for updates, this is the default. Pages are written out at, or before phase 2 commit, or at the end of an abend for transactions that have rolled back.

- **PH1** Pages are written out at, or before phase 1 commit. If a transaction subsequently rolls back, the pages are updated in the group buffer pool at the end of the rollback and are written out at the end of the abend.

- **YES** Pages are written out to the coupling facility as soon as the buffer update commits. Pages are written out regardless of whether the update occurs during forward progress or rollback of the transaction.

  This option can affect performance due to coupling facility overhead.

Install parameter IMMEDIATE WRITE on panel DSNTIP8, or ZPARM IMMEDWRI in DSN6GRP.

**Field Name:** QWP4IMMW
REBIND PLANNMGMT DEFAULT (PLANMGMT)

Controls which queries are populated in the access path repository (ZPARM parameter PLANMGMTSCOPE). Possible values are:

- **A** ALL: Includes static and dynamic SQL queries.
- **S** STATIC: Includes static SQL queries only. This is the default.
- **D** DYNAMIC: Includes dynamic SQL queries only.

**Field Name:** QWP4PMSC

PLANMGMTSCOPE DEFAULT (PLANMGMTSCOPE)

Shows if and how access path information is stored in the repository. Possible values are:

- **O** On
- **F** Off
- **B** Basic
- **E** Extended

**Field Name:** QWP4PMGT

PACKAGE RELEASE COMMIT (PKGREL_COMMIT)

YES indicates that the following operations on a package that are bound with RELEASE(DEALLOCATE) are permitted while the package is active and allocated by DB2:

- BIND and REBIND requests, including AUTOMATIC REBIND
- Data definition language changes to objects that are statically referenced by the package

The ZPARM name is PKGREL_COMMIT in DSN6SPRM.

**Field Name:** QWP4PKRC

RANDOMIZE XML DOCID (XML_RANDOMIZE_DOCID)

Specifies whether DB2 generates document ID elements sequentially or randomly. Possible values are:

- **YES** Sequentially
- **NO** Randomly

ZPARM name XML_RANDOMIZE_DOCID in DSN6SYSP.

**Field Name:** QWP1XRDI

BLOCK OPT 1 ROW SORT (OPT1ROWBLOCKSORT)

YES indicates that sort access paths are disabled (if possible) if OPTIMIZE for 1 row with a query is used. If a possible access path avoids a SORT, DB2 chooses this access path.

ZPARM name is OPT1ROWBLOCKSORT in DSN6SPRM.

**Field Name:** QWP4O1RBS

CURRENT DEGREE (CDSSRDEF)

Shows the default for the CURRENT DEGREE special register when no degree is explicitly set with SET CURRENT DEGREE. The default disables query parallelism.
Performance and Optimization (DSNTIP8, DSNTIP81)

Install parameter CURRENT DEGREE on panel DSNTIP8, or ZPARM CDSSRDEF in DSN6SPRM.

Field Name: QWP4CDEG

MAX DEGREE OF PARALLELISM (PARAMDEG)

Indicates the upper limit on the degree of parallelism for a parallel group.

This field has a value of 0. This means PARAMDEG is not set and DB2 can set a default maximum degree of parallelism based on the system configuration.

Install parameter MAX DEGREE on panel DSNTIP8, or ZPARM PARAMDEG in DSN6SPRM.

Field Name: QWP4MDEG

MAX DEGREE FOR DPSI (PARAMDEG_DPSI)

The maximum degree of parallelism for a parallel group in which a data-partitioned secondary index is used to control parallelism. This field corresponds to field MAX DEGREE FOR DPSI on installation panel DSNTIP81. The ZPARM name is PARAMDEG_DPSI in DSN6SPRM.

Field Name: QWP4DEGD

STAR JOIN ENABLING (STARJOIN)

Star join enable indicator. Possible values are:

-1 (DISABLE)
  Star join is disabled. This is the default.

0 (ENABLE)
  Star join is enabled when the join meets the conditions described in the DB2 administration information for performance.

1
  Star join is enabled without comparing the ratio of the fact-table cardinality to the cardinality of the largest dimension table. The table with the largest cardinality is the fact table.

n
  This is the star join fact table and the largest dimension table ratio. The lowest ratio of the cardinality of the fact table compared to the cardinality of the largest dimension table for which star join is used. 2 < N <= 32768.

Install parameter STAR JOIN QUERIES on panel DSNTIP8, or ZPARM STARJOIN in DSN6SPRM.

Background and Tuning Information

This parameter allows you to set the star join ratio to increase or decrease the dimension table and fact table ratio rule according to application needs.

This parameter also allows you to disable star join if needed for performance reasons. The default is to allow star join if star join detection is successful.

Star join technique is only used when these conditions exist:

- At least two dimensions exist.
- The join predicates are between the fact table and the dimension tables only. (No join predicates lie between the dimension tables.)
- The join predicates are equijoin predicates.
- No correlated subqueries cross dimensions.
No cycles within the dimensions exist. This means that no predicate can reference more than one candidate dimension table with respect to the same column of the fact table.

No outer join exists.

The data type and length of the join predicates are the same.

The fact table is larger than the dimension table.

Field Name: QWP4SJRT

MAX DATA CACHING IN MB (MXDTCACH)

The maximum amount of virtual memory in megabytes (MB) that is allocated for data caching.

Install parameter MAX DATA CACHING on panel DSNTIP8, or ZPARM MXDTCACH in DSN6SPRM.

Field Name: QWP4MXDC

CURRENT REFRESH AGE (REFSHAGE)

Shows the default for the CURRENT REFRESH AGE special register deferred materialized query tables.

Install parameter CURRENT REFRESH AGE on panel DSNTIP8, or ZPARM REFSHAGE in DSN6SPRM.

Field Name: QWP4RFSH

CURRENT MAINT TYPE (MAINTYPE)

Shows the default special register for the CURRENT MAINTAINED TABLE TYPES FOR OPTIMIZATION statement when no value is explicitly set. Possible values are:

- ALL
- NONE
- SYSTEM (default)
- USER

The default allows query rewrite using system-maintained materialized query tables (SYSTEM) when CURRENT REFRESH AGE is set to ANY. When USER, query rewrite is done using user-maintained materialized query tables when CURRENT REFRESH AGE is set to ANY. ALL means that query rewrite uses both system-maintained and user-maintained materialized query tables.

Install parameter CURRENT MAINT TYPES on panel DSNTIP8, or ZPARM MAINTYPE in DSN6SPRM.

Field Name: QWP4MNTY

UPDATE PART KEY COLS (PARTKEYU)

Indicates whether values in columns that participate in partitioning keys can be updated.

Possible values are YES, NO, or SAME. When SAME, updates are allowed only when the updated row remains in the same partition. The default value is YES.

Install parameter UPDATE PART KEY COLS on panel DSNTIP8, or ZPARM PARTKEYU in DSN6SPRM.

Field Name: QWP4PKYU
EDM BEST FIT (EDMBFIT)

Shows the free chain search algorithm on systems with a large EDM pool (greater than 40 MB). Possible values are:

YES  Use a better fit algorithm.
NO   Use a first fit algorithm.

Install parameter LARGE EDM BETTER FIT on panel DSNTIP8, or ZPARM EDMFIT in DSN6SPRM.

Field Name: QWP4EBF

VARCHAR FROM INDEX (RETVLCFK)

Indicates whether the VARCHAR column is retrieved from the index.
The data sharing scope of this parameter is GROUP.
When NO, index-only access of variable length column data is disabled.
DB2 must retrieve data from the data page. Data is retrieved with no padding.
When YES, index-only access of variable length column data is enabled.
This can improve performance. Data retrieved from the index is padded with blanks to the maximum length of the column.
Install parameter VARCHAR FROM INDEX on panel DSNTIP8, or ZPARM RETVLCFK in DSN6SPRM.

Field Name: QWP4VCFK
Protection Installation Parameters (DSNTIPP)

This topic shows detailed information about “System Parameters - Protection Installation Parameters (DSNTIPP)”.

This block shows security settings.

Data sets, including data sets defined to DFSMS, should be protected by a security manager, such as RACF.

Fields in this block can contain long names. When a long name exceeds the available space, it is truncated, the parameter identifier and the full name are printed in a separate list at the end of the report.

System Parameters - Protection Installation Parameters (DSNTIPP)

The field labels shown in the following sample layout of “System Parameters - Protection Installation Parameters (DSNTIPP)” are described in the following section.

PROTECTION INSTALLATION PARAMETERS (DSNTIPP)
--------------------------------------------
ARCHIVE LOG RACF PROTECTION (PROTECT)..................NO
DB2 AUTHORIZATION ENABLED (AUTH)..........................YES
PLAN AUTHORIZATION CACHE SIZE (AUTHCACH)..............3,072
PACKAGE AUTHORIZATION CACHE SIZE (CACHEPAC).........5,242,880
ROUTINE AUTHORIZATION CACHE SIZE (CACHERAC).........5,242,880
AUTH EXIT CHECK (AUTHEXIT_CHECK)..........................PRIMARY
AUTH EXIT CACHE REFRESH (AUTHEXIT_CACHEREFRESH)........NONE
SYSTEM ADMINISTRATOR 1 AUTHORIZATION ID (SYSADM).......HELM
SYSTEM ADMINISTRATOR 2 AUTHORIZATION ID (SYSADM2).....SYSADM
SYSTEM OPERATOR 1 AUTHORIZATION ID (SYSOPR1)..........HELM
SYSTEM OPERATOR 2 AUTHORIZATION ID (SYSOPR2)..........EMIL
DEFAULT (UNKNOWN) USER AUTHORIZATION ID (DEFLTID)......IBMUSER
RESOURCE LIMIT TABLE CREATOR AUTH ID (RLFAUTH)........SYSIBM
BIND NEW PACKAGE (BINDNV)..................................BINDADD
DBA CREATE VIEW (DBACRVW)..................................NO

ARCHIVE LOG RACF PROTECTION (PROTECT)

Indicates whether archive log data sets are protected with individual RACF profiles when they are created.

When YES, RACF protection must be active for DB2. YES also means that you cannot use RACF generic profiles for archive log data sets. If your archive log is on tape, RACF class TAPEVOL must be active, otherwise, the off-load will fail.

Install parameter ARCHIVE LOG RACF on panel DSNTIPP, or ZPARM PROTECT in DSN6ARVP.

Field Name: QWP3RTCT

DB2 AUTHORIZATION ENABLED (AUTH)

Shows whether DB2 performs authorization checking.

When all authorization checking by DB2 is disabled, the GRANT statement is also disabled (granting every privilege to PUBLIC); this is not recommended.

Install parameter USE PROTECTION on panel DSNTIPP, or ZPARM AUTH in DSN6SPRM.
Protection Installation Parameters (DSNTIPP)

Field Name: QWP4AUTH

PLAN AUTHORIZATION CACHE SIZE (AUTHCACH)

The size of the authorization cache to be used if no CACHESIZE is specified on the BIND PLAN subcommand.

The size of the cache is 32 bytes of overhead + (8 bytes of storage X number of concurrent users).

0 means authorization caching is not used.

Install parameter PLAN AUTH CACHE on panel DSNTIPP, or ZPARM AUTHCACH in DSN6SPRM.

Field Name: QWP4AUCA

PACKAGE AUTHORIZATION CACHE SIZE (CACHEPAC)

The amount of storage allocated for caching authorization information for all packages on this DB2 member.

32 KB hold about 375 collection-ID.package-IDs. The cache is stored in the DSN1DBM1 address space.

Install parameter PACKAGE AUTH CACHE on panel DSNTIPP, or ZPARM CACHEPAC in DSN6SPRM.

Field Name: QWP4PAC

ROUTINE AUTHORIZATION CACHE SIZE (CACHERAC)

The amount of storage allocated for caching authorization information for all routines on this DB2 member.

Routines include stored procedures and user-defined functions.

32 KB hold about 380 schema.routine.type entries.

Install parameter ROUTINE AUTH CACHE on panel DSNTIPP, or ZPARM CACHERAC in DSN6SPRM.

Field Name: QWP4RAC

AUTH EXIT CHECK (AUTHEXIT_CHECK)

Specifies whether the DB2 authorization ID or the RACF primary authorization ID is to be used for authorization checks, when the access control authorization exit is active:

Primary

DB2 provides:

- The ACEE of the package owner to perform statement authorization checks during AUTOMATIC REBIND, BIND, and REBIND processing
- The ACEE of the package owner, routine definer, or routine invoker, as determined by the dynamic rules behavior for dynamic SQL authorization checking, when a DYNAMICRULES BIND option value other than run is in effect.

The access control authorization exit uses the ACEE for the XAPLUCHK authorization ID field to perform the authorization.

The authorization ID in XAPLUCHK must be defined as a RACF user and must have the privileges required to execute the SQL statements in the package.
DB2 provides the ACEE of the primary authorization ID for performing all authorization checks. The primary authorization ID must have the privileges required to execute the SQL statements in the package. This field corresponds to field "RACF AUTH CHECK" on installation panel DSNTIPP. ZPARM name is RACF_AUTHCHECK in DSN6SPRM.

Field Name: QWP4RACK

AUTH EXIT CACHE REFRESH (AUTHEXIT_CACHEREFRESH)

Specifies whether the package authorization cache, routine authorization cache, and dynamic statement cache entries are refreshed when an access control authorization exit is active, and the user profile is changed in RACF. Possible values are:

• All
• None

This field corresponds to field AUTH EXIT CACHE REFR in installation panel DSNTIPP. ZPARM name is AUTHEXIT_CACHEREFRESH in DSN6SPRM.

Field Name: QWP4AECR

SYSTEM ADMINISTRATOR 1 AUTHORIZATION ID (SYSADM)

One of two authorization IDs with SYSADM authority. SYSADM users can access to DB2 in all cases.

This identifier can be a long string. If there is insufficient space to show the complete string, the string is truncated in the report block. The complete string is shown in a separate list of long names at the end of the report.

Install parameter SYSTEM ADMIN 1 on panel DSNTIPP, or ZPARM SYSADM in DSN6SPRM.

Field Name: QWP4ADM

SYSTEM ADMINISTRATOR 2 AUTHORIZATION ID (SYSADM2)

One of two authorization IDs with SYSADM authority. SYSADM users can access to DB2 in all cases.

This identifier can be a long string. If there is insufficient space to show the complete string, the string is truncated in the report block. The complete string is shown in a separate list of long names at the end of the report.

Install parameter SYSTEM ADMIN 2 on panel DSNTIPP, or ZPARM SYSADM2 in DSN6SPRM.

Field Name: QWP4ADM2

SYSTEM OPERATOR 1 AUTHORIZATION ID (SYSOPR1)

One of two authorization IDs with SYSOPR authority. SYSOPR users can access DB2 even if the DB2 catalog is unavailable.

This identifier can be a long string. If there is insufficient space to show the complete string, the string is truncated in the report block. The complete string is shown in a separate list of long names at the end of the report.

Install parameter SYSTEM OPERATOR 1 on panel DSNTIPP, or ZPARM SYSOPR1 in DSN6SPRM.

Field Name: QWP4OPR1
SYSTEM OPERATOR 2 AUTHORIZATION ID (SYSOPR2)

One of two authorization IDs with SYSOPR authority. SYSOPR users can access DB2 even if the DB2 catalog is unavailable.

This identifier can be a long string. If there is insufficient space to show the complete string, the string is truncated in the report block. The complete string is shown in a separate list of long names at the end of the report.

Install parameter SYSTEM OPERATOR 2 on panel DSNTIPP, or ZPARM SYSOPR2 in DSN6SPRM.

Field Name: QWP4OPR2

DEFAULT (UNKNOWN) USER AUTHORIZATION ID (DEFLTID)

The authorization ID used if RACF is not available for batch access and USER= is not specified in the job statement.

This identifier can be a long string. If there is insufficient space to show the complete string, the string is truncated in the report block. The complete string is shown in a separate list of long names at the end of the report.

Install parameter UNKNOWN AUTHID on panel DSNTIPP, or ZPARM DEFLTID in DSN6SPRM.

Field Name: QWP4DFID

RESOURCE LIMIT TABLE CREATOR AUTH ID (RLFAUTH)

The authorization ID used for the resource limit facility (governor).

This identifier can be a long string. If there is insufficient space to show the complete string, the string is truncated in the report block. The complete string is shown in a separate list of long names at the end of the report.

Install parameter RESOURCE AUTHID on panel DSNTIPP, or ZPARM RLFAUTH in DSN6SYSP.

Field Name: QWP1RLFA

BIND NEW PACKAGE (BINDNV)

Shows whether BIND or BINDADD authority is required to BIND a new version of an existing package.

When BINDADD (default), only users with BINDADD system privilege can create a new package.

BIND users with BIND privilege for a package or collection can create a new version of an existing package when they bind it. This also allows users with PACKADM authority to add a new package or a new version of a package to a collection.

Install parameter BIND NEW PACKAGE on panel DSNTIPP, or ZPARM BINDNV in DSN6SPRM.

Field Name: QWP4BNVA

DBA CREATE VIEW (DBACRVW)

Shows whether a DB2 administrator can create a view or alias for another user. Possible values are YES or NO. The default is NO.

Install parameter DBADM CREATE AUTH on panel DSNTIPP, ZPARM DBACRVW in macro DSN6SPRM.

Field Name: QWP4CRVW
Protection Panel (DSNTIPP1)

This topic shows detailed information about “System Parameters - Protection Panel (DSNTIPP1)”.

System Parameters - Protection Panel (DSNTIPP1)

The field labels shown in the following sample layout of “System Parameters - Protection Panel (DSNTIPP1)” are described in the following section.

PROTECTION PANEL (DSNTIPP1)

---------------------------
SECURITY ADMINISTRATOR 1 AUTHORIZATION ID (SECADM1)......SECADM
SECURITY ADMINISTRATOR 1 TYPE (SECADM1_TYPE).............AUTHID
SECURITY ADMINISTRATOR 2 AUTHORIZATION ID (SECADM2)......SECADM
SECURITY ADMINISTRATOR 2 TYPE (SECADM2_TYPE).............AUTHID
SEPARATE SECURITY DUTIES (SEPARATE_SECURITY)...............NO
INCLUDE DEPENDENT PRIVILEGES (REVOKE_DEP_PRIVILEGES)........N

SECURITY ADMINISTRATOR 1 AUTHORIZATION ID (SECADM1)

Security administrator 1 authorization ID (blank if ROLE).

This field corresponds to field SECURITY ADMIN 1 on installation panel DSNTIPP1, or ZPARM SECADM1 in DSN6SPRM.

Field Name: QWP4SECA1_E

SECURITY ADMINISTRATOR 1 TYPE (SECADM1_TYPE)

Security administrator 1 type. Possible values are:

' ' Indicates that the authorization ID (AUTH ID) is used.
'L' Indicates that ROLE is used.

This field corresponds to field SEC ADMIN 1 TYPE on installation panel DSNTIPP1, or ZPARM SECADM1_TYPE in DSN6SPRM.

Field Name: QWP4SECA1_TYPE

SECURITY ADMINISTRATOR 2 AUTHORIZATION ID (SECADM2)

Security administrator 2 authorization ID (blank if ROLE).

This field corresponds to field SECURITY ADMIN 2 on installation panel DSNTIPP1, or ZPARM SECADM2 in DSN6SPRM.

Field Name: QWP4SECA2_E

SECURITY ADMINISTRATOR 2 TYPE (SECADM2_TYPE)

Security administrator 2 type. Possible values are:

'blank' Indicates that the authorization ID (AUTH ID) is used.
'L' Indicates that ROLE is used.

This field corresponds to field SEC ADMIN 2 TYPE on installation panel DSNTIPP1, or ZPARM SECADM2_TYPE in DSN6SPRM.

Field Name: QWP4SECA2_TYPE

SEPARATE SECURITY DUTIES (SEPARATE_SECURITY)

Separate security tasks. Possible values are:

Y SYSADM/SYSCTRL cannot GRANT/REVOKE
N SYSADM/SYSCTRL can GRANT/REVOKE
Protection Panel (DSNTIPP1)

Field Name: QWP4SEPSD

INCLUDE DEPENDENT PRIVILEGES (REVOKE_DEP_PRIVILEGES)

Include dependent privileges on REVOKE. Possible values are:

Y  If INCLUDING DEPENDENT PRIVILEGES is enforced.
N  If NOT INCLUDING DEPENDENT PRIVILEGES is enforced.
S  If specified in a REVOKE statement.

Field Name: QWP4RVDPR
This topic shows detailed information about “System Parameters - Query Accelerator Preferences (DSNTIP82)”.

**System Parameters - Query Accelerator Preferences (DSNTIP82)**

The field labels shown in the following sample layout of “System Parameters - Query Accelerator Preferences (DSNTIP82)” are described in the following section.

**ACCELERATOR STARTUP OPTION (ACCEL)**

Specifies whether to enable accelerator servers. Possible values are:

- **AUTO**
  - Enable and start accelerator servers.

- **COMMAND**
  - Enable but do not start accelerator servers.

- **NO**
  - Do not enable accelerator servers.

This field corresponds to field ACCEL STARTUP on installation panel DSNTIP81. ZPARM name is ACCEL in DSN6SPRM.

**CURRENT GET ACCEL ARCHIVE (GET_ACCEL_ARCHIVE)**

Determines the default value that is to be used for the CURRENT GET_ACCEL_ARCHIVE special register:

- **NO**
  - Indicates that if a table is archived in an accelerator server, and a query references that table, the query does not use the data that is archived.

- **YES**
  - Indicates that if a table is archived in an accelerator server, and a query references that table, the query uses the data that is archived.

ZPARM name GET_ACCEL_ARCHIVE in macro DSN6SPRM.

**ACCELERATION OPTIONS (QUERY_ACCEL_OPTIONS)**

Specifies additional types of SQL queries that are eligible for acceleration.

- **NONE**
  - Indicates that no additional types of SQL queries are eligible. Therefore, the types of queries that are described in the other available values for this parameter are not eligible for acceleration. This is the default value.

- **1**
  - Indicates that queries that include data that is encoded with the EBCDIC mixed or graphic encoding schemes are eligible for acceleration.

- **2**
  - Indicates that an INSERT with SELECT statement is eligible for acceleration.
Query Accelerator Preferences (DSNTIP82)

acceleration. However, only the SELECT operation of the query is processed by the accelerator server.

3 Indicates that queries that contain built-in functions for which DB2 processes each byte of the input string, rather than each character of the input string, can run on an accelerator server.

4 The queries that reference an expression with a DATE data type that uses a LOCAL format are not blocked from executing on IDAA. IDAA will use the dd/mm/yyyy format to interpret the input and output date value. Specify option 4 only when you also specify LOCAL as the setting for the DSNHDECP.DATE parameter and your LOCAL date exit defines the specific dd/mm/yyyy date format. Otherwise, queries may return unpredictable results.

ZPARM name QUERY_ACCEL_OPTIONS in macro DSN6SPRM.

Field Name: QWP4QACO

CURRENT QUERY ACCEL (QUERY_ACCELERATION)

Determines the default value that is to be used for the CURRENT QUERY ACCELERATION special register. Possible values are:

NONE
Indicates that no query acceleration is done. This is the default value.

ENABLE
Indicates that queries are accelerated only if DB2 determines that it is advantageous to do so. If there is an accelerator failure while a query is running, or the accelerator returns an error, DB2 returns a negative SQLCODE to the application.

ENABLE_WITH_FAILBACK
Indicates that queries are accelerated only if DB2 determines that it is advantageous to do so. If the accelerator returns an error during the PREPARE or first OPEN for the query, DB2 executes the query without the accelerator. If the accelerator returns an error during a FETCH or a subsequent OPEN, DB2 returns the error to the user, and does not execute the query.

ELIGIBLE
Indicates that queries are accelerated if they are eligible for acceleration. DB2 does not use cost information to determine whether to accelerate the queries. Queries that are not eligible for acceleration are executed by DB2. If there is an accelerator failure while a query is running, or the accelerator returns an error, DB2 returns a negative SQLCODE to the application.

ALL
Indicates that queries are accelerated if they are eligible for acceleration. DB2 does not use cost information to determine whether to accelerate the queries. Queries that are not eligible for acceleration are not executed by DB2, and an SQL error is returned. If there is an accelerator failure while a query is running, or the accelerator returns an error, DB2 returns a negative SQLCODE to the application.

ZPARM name QUERY_ACCELERATION in DSN6SPRM.

Field Name: QWP4CQAC
Routine Parameters (DSNTIPX)

This topic shows detailed information about “System Parameters - Routine Parameters (DSNTIPX)”. This block shows information about the stored procedures address space used to run stored procedures or user-defined functions.

**Note:** The fields shown on this panel depend on the installed DB2 version.

**System Parameters - Routine Parameters (DSNTIPX)**

The field labels shown in the following sample layout of “System Parameters - Routine Parameters (DSNTIPX)” are described in the following section.

```
Routine Parameters (DSNTIPX)
--------------------
MAX ABEND COUNT (STORMXAB)............................0
TIMEOUT VALUE (STORTIME)................................180
WLM ENVIRONMENT (WLMENV).............................WLMENV
MAX OPEN CURSORS (MAX_NUM_CUR)........................500
MAX STORED PROCS (MAX_ST_PROC).....................2,000
```

**MAX ABEND COUNT (STORMXAB)**

The number of times a stored procedure is allowed to terminate abnormally, after which SQL CALL statements for the stored procedure are rejected.

Install parameter MAX ABEND COUNT on panel DSNTIPX, or ZPARM STORMXAB in DSN6SYSP.

**Field Name:** QWP1SPAB

**TIMEOUT VALUE (STORTIME)**

The number of seconds before DB2 stops waiting for an SQL CALL statement to be assigned to one of the TCBs in the DB2 stored procedures address space.

Install parameter TIMEOUT VALUE on panel DSNTIPX, or ZPARM STORTIME in DSN6SYSP.

**Field Name:** QWP1SPTO

**WLM ENVIRONMENT (WLMENV)**

Workload manager environment.

Install parameter WLM ENVIRONMENT on panel DSNTIPX, or ZPARM WLMENV in DSN6SYSP.

**Field Name:** QWP1WLME

**MAX OPEN CURSORS (MAX_NUM_CUR)**

Shows the maximum number of cursors, including allocated cursors, that are open at a given DB2 site per thread. RD5 keeps a total of currently open cursors. If an application attempts to open a thread after the maximum is reached, the statement will fail.

In a data sharing group, this parameter is shown at member scope.

Install parameter MAX OPEN CURSORS on panel DSNTIPX, or ZPARM MAX_NUM_CUR in DSN6SPRM.
Routine Parameters (DSNTIPX)

Field Name: QWP4MXNC
MAX STORED PROCS (MAX_ST_PROC)

Shows the maximum number of stored procedures per thread. If an application attempts to call a stored procedure after this is reached, the statement will fail. In a data sharing group, this parameter is shown as member scope.

Install parameter MAX STORED PROCS on panel DSNTIPX, or ZPARM MAX_ST_PROC in DSN6SPRM.

Field Name: QWP4MXSP
This topic shows detailed information about “System Parameters - Sizes Panel 1 (DSNTIPD)”.

System Parameters - Sizes Panel 1 (DSNTIPD)

The field labels shown in the following sample layout of “System Parameters - Sizes Panel 1 (DSNTIPD)” are described in the following section.

Sizes Panel 1 (DSNTIPD)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>USER LOB VALUE STORAGE IN KB (LOBVALA)</td>
<td>10,240</td>
</tr>
<tr>
<td>SYSTEM LOB VALUE STORAGE IN MB (LOBVALS)</td>
<td>2,048</td>
</tr>
<tr>
<td>MAXIMUM NUMBER OF LE TOKENS (LEMAX)</td>
<td>20</td>
</tr>
<tr>
<td>USER XML VALUE STG IN KB (XMLVALA)</td>
<td>11</td>
</tr>
<tr>
<td>SYSTEM XML VAL STG IN MB (XMLVALS)</td>
<td>12</td>
</tr>
</tbody>
</table>

USER LOB VALUE STORAGE IN KB (LOBVALA)

The maximum amount of storage (KB) each user can use for LOB values.

Install parameter USER LOB VALUE STORAGE on panel DSNTIP7, or ZPARM LOBVALA in DSN6SYSP.

Field Name: QWP1LVA

SYSTEM LOB VALUE STORAGE IN MB (LOBVALS)

The maximum amount of storage (MB) each system can use for LOB values.

Install parameter SYSTEM LOB VALUE STORAGE on panel DSNTIP7, or ZPARM LOBVALS in DSN6SYSP.

Field Name: QWP1LVS

MAXIMUM NUMBER OF LE TOKENS (LEMAX)

The maximum number of LE tokens active at any time. When zero, no tokens are available.

A token is used each time one of the following is used: trigonometry functions, degrees, radians, rand, exp, power, log functions, upper, lower, translate.

Install parameter MAXIMUM LE TOKENS on panel DSNTIP7, or ZPARM LEMAX in DSN6SPRM.

Field Name: QWP4LEM

USER XML VALUE STG IN KB (XMLVALA)

The maximum amount of memory in kilobytes (KB) for each user for storing XML values.

ZPARM XMLVALA in DSN6SYSP.

Field Name: QWP1XVA

SYSTEM XML VAL STG IN MB (XMLVALS)

The maximum amount of memory in megabytes (MB) for each system for storing XML values.

ZPARM XMLVALS in DSN6SYSP.

Field Name: QWP1XVS
SQL Object Defaults Panel (DSNTIP7, DSNTIP71)

This topic shows detailed information about “System Parameters - SQL Object Defaults Panel (DSNTIP7, DSNTIP71)”.

This block shows the limits for the amount of storage that can be used for storing large object (LOB) values.

Note: The fields shown on this panel depend on the installed DB2 version.

System Parameters - SQL Object Defaults Panel (DSNTIP7, DSNTIP71)

The field labels shown in the following sample layout of “System Parameters - SQL Object Defaults Panel (DSNTIP7, DSNTIP71)” are described in the following section.

SQL OBJECT DEFAULTS PANEL (DSNTIP7,DSNTIP71)
--------------------------------------------
OBJECT CREATE FORMAT (OBJECT_CREATE_FORMAT)............EXTENDED
UTILITY OBJECT CONVERSION (UTILITY_OBJECT_CONVERSION)......NONE
VARY DS CONTROL INTERVAL (DSVCI)............................YES
TABLE SPACE ALLOCATION IN KB (TSQTY)..........................0
INDEX SPACE ALLOCATION IN KB (IXQTY)..........................0
OPTIMIZE EXTENT SIZING (MGEXTSZ)............................NO
PAD INDEX BY DEFAULT (PADIX)...............................NO
DEFAULT PARTITION SEGSIZE (DPSEG SZ)........................32
PERCENT FREE FOR UPDATE (PCTFREE_UPD)......................0
DEFINE DATA SETS (IMPSDSDEF).................................YES
USE DATA COMPRESSION (IMPTSCMP)..............................NO
LIMIT KEY CONV PART TAB (IX_TB_PART_CONV_EXCLUDE).........NO
LOB INLINE LENGTH (LOB_INLINE_LENGTH).......................0

OBJECT CREATE FORMAT (OBJECT_CREATE_FORMAT)

Creates new table spaces and indexes in the following log record format:

EXTENDED

Creates new table spaces and indexes in extended log record format.

BASIC

Creates new table spaces and indexes in basic log record format.

Field Name: QWP4OBCF

UTILITY OBJECT CONVERSION (UTILITY_OBJECT_CONVERSION)

This field can have the following values:

NONE (QWP4UTOC1=0 and QWP4UTOC2=0)

No conversion is performed. This option is the default setting of this parameter. NONE is allowed regardless of the OBJECT CREATE FORMAT setting.

BASIC (QWP4UTOC1=1 and QWP4UTOC2=0)

Existing table spaces and indexes that use extended 10-byte page format are converted to basic 6-byte page format. BASIC is allowed only if the OBJECT CREATE FORMAT field is also set to BASIC.

EXTENDED (QWP4UTOC1=0 and QWP4UTOC2=1)

Existing table spaces and indexes that use 6-byte page format are
SQL Object Defaults Panel (DSNTIP7, DSNTIP71)

converted to extended 10-byte page format. EXTENDED is allowed
only if the OBJECT CREATE FORMAT field is also set to
EXTENDED.

NO BASIC (QWP4UTOC1=1 and QWP4UTOC2=1)
Prevents the conversion of table spaces and indexes in extended
page format to basic page format and disallows a utility that
accepts the RBALRNS_CONVERSION utility keyword from
running on an object in basic page format unless it converts it to
extended page format. This setting is permitted only when
OBJECT_CREATE_FORMAT=EXTENDED is set.
The ZPARM name is UTILITY_OBJECT_CONVERSION in DSN6SPRM.

Field Name: RT0106OC

VARY DS CONTROL INTERVAL (DSVCI)
Indicates whether DB2 optimizes VSAM CONTROL INTERVAL to page
size for data set allocation.
Install parameter VARY DS CONTROL INTERVAL on panel DSNTIP7, or
ZPARM DSVCI in DSN6SYSP.

Field Name: QWP1VVCI

TABLE SPACE ALLOCATION IN KB (TSQTY)
Shows the amount of space in KB for primary and secondary space
allocation for DB2-defined data sets for table spaces created without the
USING clause. 0 indicates that DB2 uses standard defaults.
Install parameter TABLE SPACE ALLOCATION on panel DSNTIP7, or
ZPARM TSQTY in DSN6SYSP.

Field Name: QWP1TSQT

INDEX SPACE ALLOCATION IN KB (IXQTY)
Shows the amount of space in KB for primary and secondary space
allocation for DB2-defined data sets for index spaces created without the
USING clause. 0 indicates that DB2 uses standard defaults.
Install parameter INDEX SPACE ALLOCATION on panel DSNTIP7, or
ZPARM IXQTY in DSN6SYSP.

Field Name: QWP1IXQT

OPTIMIZE EXTENT SIZING (MGEXTSZ)
Indicates whether DB2 uses sliding secondary quantity for DB2 managed
data sets to optimize extent sizing.
Install parameter OPTIMIZE EXTENT SIZING on panel DSNTIP7, or
ZPARM MGEXTSZ in DSN6SYSP.

Field Name: QWP1MESZ

PAD INDEX BY DEFAULT (PADIX)
Shows whether new indexes are be padded by default.
• YES indicates that a new index is padded unless the NOT PADDED
  option is specified on the CREATE INDEX statement.
• The default value, NO, indicates that a new index is not padded unless
  the PADDED option is specified on the CREATE INDEX statement.
SQL Object Defaults Panel (DSNTIP7, DSNTIP71)

Install parameter PAD INDEXES BY DEFAULT on installation panel DSNTIPE, or ZPARM PADIX in DSN6SPRM.

Field Name: QWP4PDIX

DEFAULT PARTITION SEGSIZE (DPSEGSZ)

The default segment size to be used for a partitioned table space when the CREATE TABLESPACE statement does not include the SEGSIZE parameter. This field corresponds to field DEFAULT PARTITION SEGSIZE on installation panel DSNTIP7. The ZPARM name is DPSEGSZ in DSN6SYSP.

Field Name: QWP1DPSS

PERCENT FREE FOR UPDATE (PCTFREE_UPD)

Specifies the default percentage of each page that DB2 leaves as free space in a table space when a table in this table space is populated. This value applies only to table spaces whose definitions do not include PCTFREE and for UPDATE.

This value corresponds to field PERCENT FREE FOR UPDATE on installation panel DSNTIP71. The ZPARM name is PCTFREE_UPD in DSN6SPRM.

Field Name: QWP4PFUP

DEFINE DATA SETS (IMPDSDEF)

Defines the underlying data sets when a table space (TS) that is contained in an implicitly created database is created.

Install parameter DEFINE DATA SETS on panel DSNTIP7 or ZPARM IMPDSDEF in DSN6SYSP.

Field Name: QWP1DIDS

USE DATA COMPRESSION (IMPTSCMP)

Shows whether data compression in table spaces in implicitly defined databases is used.

Install parameter USE DATA COMPRESSION on panel DSNTIP7 or ZPARM IMPTSCMP in DSN6SYSP.

Field Name: QWP1CITS

LIMIT KEY CONV PART TAB (IX_TPB_PART_CONV_EXCLUDE)

Shows whether to include all columns in the partitioning key during conversion from index-controlled partitioning to table-controlled partitioning:

NO  Includes all columns
YES  Includes trailing columns only if they affect partitioning

This field corresponds to field EXCLUDE PART KEY ELEMENTS on installation panel DSNTIP71. The ZPARM name is IX_TPB_PART_CONV_EXCLUDE in DSN6SPRM.

Field Name: QWP4XPKE

LOB INLINE LENGTH (LOB_INLINE_LENGTH)

Default inline length for any new storing large object (LOB) column in a Universal Table Space on the DB2 subsystem. The valid values are from 0 to 32680 inclusive (in bytes). The default value for this ZPARM is 0, which
indicates that no inline attribute is desired for any LOB column (BLOB, CLOB or DBCLOB) created on this subsystem.

**Field Name:** QWP1LBIL
This topic shows detailed information about “System Parameters - Storage Sizes Installation Parms (DSNTIPC, DSNTIPE, DSNTIPE1)”.

This block shows the storage sizes calculated by the installation CLIST.

These space estimates do not account for cylinder rounding. Base requirements can be 10 to 20% higher depending on the DASD type. Most of the needed virtual storage is in extended private storage (including the buffer pool, the EDM pool, most of the code, and a significant amount of working storage).

Note: The fields shown on this panel depend on the installed DB2 version.

### System Parameters - Storage Sizes Installation Parms (DSNTIPC, DSNTIPE, DSNTIPE1)

The field labels shown in the following sample layout of “System Parameters - Storage Sizes Installation Parms (DSNTIPC, DSNTIPE, DSNTIPE1)” are described in the following section.

```
STORAGE SIZES INSTALLATION PARMS (DSNTIPC,DSNTIPE,DSNTIPE1)
-----------------------------------------------
MAX NO OF DATA SETS CONCURRENTLY IN USE (DSMAX)........20,000
EDM STATEMENT CACHE SIZE IN KB (EDMSTMTC)................113,386
EDM DBD CACHE SIZE IN KB (EDMDBDC).......................102,400
EDM SKELETON POOL SIZE IN KB (EDM_SKELETON_POOL)........102,400
MAXIMUM SIZE OF EDM POOL IN BYTES (EDMPOOL)..............0
MAXIMUM SIZE OF SORT POOL IN BYTES (SORTPOOL)...........10,240,000
MAX IN-MEMORY SORT SIZE (MAXSORT_IN_MEMORY).............1,000
MAXIMUM SIZE OF RID POOL IN KB (MAXRBLK)...............400,000
MAX NO OF USERS CONCURRENTLY RUNNING IN DB2 (CTHREAD).....400
MAX NO OF CONCURRENT REMOTE ACTIVE CONNECTIONS (MAXDBAT)....200
MAX NO OF REMOTE CONNECTIONS (CNDBAT)....................10,000
MAX NO OF TSO CONNECTIONS (IDFORE)......................200
MAX NO OF BATCH CONNECTIONS (IDBACK).....................200
MAXIMUM KEPT DYNAMIC STATEMENTS (MAXKEEPD)..............5,000
MAX OPEN FILE REFS (MAXOPFR)................................100
CONTRACT THREAD STORAGE (CONSTOR).......................YES
MANAGE THREAD STORAGE (MINSTOR)........................NO
LONG-RUNNING READER IN MINUTES (LRDRTHLD)..................10
DDL TIMEOUT FACTOR (DDLTOX)...............................1
INDEX CLEANUP THREADS (INDEX_CLEANUP_THREADS)...........10
3990 CACHE (SEQCACH)..................................SEQ
```

**MAX NO OF DATA SETS CONCURRENTLY IN USE (DSMAX)**

The maximum number of data sets that can be open at one time.

The practical limit can be less than the MVS limit of 32727, depending on available storage below the line.

Install parameter DSMAX on panel DSNTIPC, or ZPARM DSMAX in DSN6SPRM.

Field Name: QWP4DSMX

**EDM STATEMENT CACHE SIZE IN KB (EDMSTMTC)**

The size of the statement cache that can be used by the Environmental Descriptor Manager (EDM). This value is used at DB2 startup time as the minimum value. You can increase and subsequently decrease this value with the SET SYSPARM command. This value cannot be decreased below
Storage Sizes Installation Parms (DSNTIPC, DSNTIPE, DSNTIPE1)

the value that is specified at DB2 startup. The CLIST calculates a statement cache size. This storage pool is located above the 2 GB bar.

The value used at DB2 startup time is either calculated by the CLIST based on input from other installation information or an override value.

For record trace, this value is shown in bytes. For other reports, the value is shown in kilobytes.

Install parameter EDM STATEMENT CACHE on panel DSNTIPC, or ZPARM EDMSTMTC in DSN6SPRM.

Field Name: QWP4ESTC

EDM DBD CACHE SIZE IN KB (EDMDBDC)

The minimum size of the DBD cache that can be used by the Environmental Descriptor Manager (EDM). This value is used at DB2 startup time as the minimum value. You can increase and subsequently decrease the value with the SET SYSPARM command. This value cannot be decreased below the value that is specified at DB2 startup. This storage pool is located above the 2 GB bar. The CLIST calculates the DBD cache size.

The value used at DB2 startup time is either calculated by the CLIST based on input from other installation information or an override value.

Install parameter EDM DBD CACHE on panel DSNTIPC, or ZPARM EDMDBDC in DSN6SPRM.

Field Name: QWP4EDBC

EDM SKELETON POOL SIZE IN KB (EDM_SKELETON_POOL)

The minimum size of the EDM pool for skeleton package and skeleton cursor tables. For record trace, this value is shown in bytes. For other reports, the value is shown in kilobytes.

Install parameter EDM SKELETON POOL SIZE on panel DSNTIPC or ZPARM EDM_SKELETON_POOL in DSN6SPRM.

Field Name: QWP4SKLC

MAXIMUM SIZE OF EDM POOL IN BYTES (EDMPOOL)

The size (in kilobytes) of the environmental descriptor manager (EDM) pool.

This can be the value calculated by the CLIST, based on input from previous panels, or the value entered in the Override column at installation time.

Install parameter EDMPOOL STORAGE SIZE on panel DSNTIPC, or ZPARM EDMPOOL in DSN6SPRM.

Field Name: QWP4EDPL

MAXIMUM SIZE OF SORT POOL IN BYTES (SRTPOOL)

Indicates the amount of storage needed for the sort pool.

This can be the value calculated by the CLIST, based on input from previous panels, or the value entered in the Override column at installation time.

Install parameter SORT POOL SIZE on panel DSNTIPC, or ZPARM SRTPOOL in DSN6SPRM.
Field Name: QWP4SPOL

MAX IN-MEMORY SORT SIZE (MAXSORT_IN_MEMORY)

The maximum amount of storage in kilobytes to allocate for sorting the results of each query that contains the order by clause, the group by clause, or both. This field corresponds to field MAX IN-MEMORY SORT SIZE in installation panel DSNTIPC. The ZPARM name is MAXSORT_IN_MEMORY in DSN6SPRM.

Field Name: QWP4MIMTS

MAXIMUM SIZE OF RID POOL IN KB (MAXRBLK)

The amount of storage needed for the RID pool. This can be the value calculated by the CLIST, based on input from previous panels, or the value entered in the Override column at installation time. When 0, DB2 does not use access paths or join methods that depend on RID pool storage.

Install parameter RID POOL SIZE on panel DSNTIPC, or ZPARM MAXRBLK in DSN6SPRM.

Field Name: QWP4RMX

MAX NO OF USERS CONCURRENTLY RUNNING IN DB2 (CTHREAD)

The maximum number of allied threads (threads started at the local subsystem) that can be allocated concurrently.

Separate threads are created for each occurrence of the following:
- TSO user (whether running a DSN command or a DB2 request from QMF)
- Batch job (whether running a DSN command or a DB2 utility)
- IMS region that can access DB2
- Active CICS transaction that can access DB2
- Task connected to DB2 through the call attachment facility.

Install parameter MAX USERS on panel DSNTIPE, or ZPARM CTHREAD in DSN6SYSP.

Field Name: QWP1CT

MAX NO OF CONCURRENT REMOTE ACTIVE CONNECTIONS (MAXDBAT)

The maximum number of database access threads (DBATs) that can be active concurrently.

When this limit has been reached, DB2 uses the value of DDF THREADS on panel DSNTIPR to decide how to handle a new allocation request.

When DDF THREADS is ACTIVE and MAX REMOTE CONNECTED has not been reached, the allocation request is allowed but any further processing for the connection is queued waiting for an active database access thread to terminate.

When DDF THREADS is INACTIVE and MAX REMOTE CONNECTED has not been reached, the allocation request is allowed and is processed when DB2 can assign an unused database access thread slot to the connection.
The total number of threads accessing data concurrently is the sum of MAX USERS and MAX REMOTE ACTIVE. The maximum allowable value for this sum is 2000.

Install parameter MAX REMOTE ACTIVE on panel DSNTIPE, or ZPARM MAXDBAT in DSN6SYSP.

**Field Name:** QWP1RMT

**MAX NO OF REMOTE CONNECTIONS (CONDBAT)**

The maximum allowed number of concurrent remote connections. When this limit is reached, any new connection request is rejected.

Install parameter MAX REMOTE CONNECTED on panel DSNTIPE, or ZPARM CONDBAT in DSN6SYSP.

**Field Name:** QWP1CDB

**MAX NO OF TSO CONNECTIONS (IDFORE)**

The maximum number of concurrent TSO foreground connections (QMF, DSN, DB2I, and SPUFI). Each of the following is a separate user:

- Each TSO foreground user executing a DSN command.
- Each TSO foreground user connected to DB2 through the call attachment facility (CAF). This can include QMF users running in TSO foreground or user-written CAF applications running in TSO foreground.

When the number of TSO users attempting to access DB2 exceeds this limit, connection requests are rejected.

There is no subsystem parameter to control the maximum concurrent connections for IMS and CICS. These are controlled by using IMS and CICS facilities. For CICS attachment, the maximum number of connections to DB2 can be controlled using the resource control table (RCT) TYPE=INIT THRDMAX value.

Install parameter MAX TSO CONNECT on panel DSNTIPE, or ZPARM IDFORE in DSN6SYSP.

**Field Name:** QWP1IDF

**MAX NO OF BATCH CONNECTIONS (IDBACK)**

The maximum allowed number of concurrent connections for batch jobs and utilities. This includes:

- All batch jobs using QMF.
- All batch jobs using the DSN command processor.
- All tasks connected to DB2 through call attach facility (CAF) running in batch. This can include:
  - Batch jobs using QMF
  - APPC applications
  - TCP/IP FTP connections

When the number of batch jobs reaches this limit, further requests are rejected.

Install parameter MAX BATCH CONNECT on panel DSNTIPE, or ZPARM IDBACK in DSN6SYSP.
Storage Sizes InstallationParms (DSNTIPC, DSNTIPE, DSNTIPE1)

Field Name: QWP1IDB

MAXIMUM KEPT DYNAMIC STATEMENTS (MAXKEEPD)
Shows the total number of prepared dynamic SQL statements that are saved past a commit point.

0 means that prepared dynamic SQL statements are not saved past commit points.

Install parameter MAX KEPT DYN STMNTS on panel DSNTIPE, or ZPARM MAXKEEPD in DSN6SPRM.

Field Name: QWP4MXKD

MAX OPEN FILE REFS (MAXOFILR)
The maximum number of concurrently open data sets for processing LOB file references.

Install parameter MAX OPEN FILE REFS on panel DSNTIPE or ZPARM MAXOFILR in DSN6SYSP.

Field Name: QWP1MOFR

CONTRACT THREAD STORAGE (CONTSTOR)
Indicates whether DB2 returns unused thread storage at commit. Possible values are:

YES  DB2 checks threads at commit points and periodically returns unused storage to the system.

NO   DB2 does not check threads at commit points and returns acquired storage on deallocation.

Install parameter CONTRACT THREAD STG on panel DSNTIPE, or ZPARM CONTSTOR in DSN6SPRM.

Field Name: QWP4CONT

MANAGE THREAD STORAGE (MINSTOR)
Shows whether DB2 uses storage management to optimize the amount of working storage consumed by individual threads.

Install parameter MANAGE THREAD STORAGE on panel DSNTIPE, or ZPARM MINSTOR in DSN6SPRM.

For best performance, this parameter should be NO, meaning DB2 does not manage thread storage.

When YES, DB2 uses best fit algorithm to manage and assign thread storage. This can help on systems that have many long-running threads and that are constrained on DBM1 address space.

Field Name: QWP4MSTG

LONG-RUNNING READER IN MINUTES (LRDRTHLD)
Shows the number of minutes that a read claim can be held by an agent before DB2 reports it as a long-running reader. Valid values are 0 (default) through 1439.

Install parameter LONG-RUNNING READER on installation panel DSNTIPE, or ZPARM LRDRTHLD in DSN6SYSP.

Field Name: QWP4LRTH
**Storage Sizes Installation Parms (DSNTIPC, DSNTIPE, DSNTIPE1)**

**DDL TIMEOUT FACTOR (DDLTOX)**

Shows the time out factor of the SQL data definition. The time out value is the product of this value and the IRLMRWT value.

ZPARM name DDLTOX in DSN6SPRM.

**Field Name:** QWP4DDLTO

**INDEX CLEANUP THREADS (INDEX_CLEANUP_THREADS)**

The maximum number of threads that can be created to clean up pseudo-deleted index entries on a data sharing member of a subsystem. This field corresponds to field INDEX CLEANUP THREADS on installation panel DSNTIPE1. The ZPARM name is INDEX_CLEANUP_THREADS in DSN6SPRM.

**Field Name:** QWP4IXCU

**3990 CACHE (SEQCACH)**

Indicates whether DB2 prefetch uses sequential mode to read cached data from a 3990 controller. When BYPASS (default), DB2 prefetch bypasses the cache.

When SEQ, DB2 prefetch uses sequential access for read activity. There is a performance benefit using SEQ with DFSMS or DFP controls with newer 3990 caches.

Install parameter SEQUENTIAL CACHE on panel DSNTIPE, or ZPARM SEQCACH in DSN6SPRM.

**Field Name:** QWP4SCAC
Tracing, Checkpoint & Pseudo-Close Parameters (DSNTIPN)

This topic shows detailed information about “System Parameters - Tracing, Checkpoint & Pseudo-Close Parameters (DSNTIPN)”.

This block shows audit, global, accounting, and monitor trace and checkpoint frequency parameters.

System Parameters - Tracing, Checkpoint & Pseudo-Close Parameters (DSNTIPN)

The field labels shown in the following sample layout of “System Parameters - Tracing, Checkpoint & Pseudo-Close Parameters (DSNTIPN)” are described in the following section.

Tracing, Checkpoint & Pseudo-Close Parameters (DSNTIPN)

```
START AUDIT TRACE (AUDITST)..................................NO
START GLOBAL TRACE (TRACSTR).................................NO
TRACE TABLE SIZE IN 4K BYTES (TRACTBL).......................16
START SMF ACCOUNTING (SMFACCT)...........................1
START SMF STATISTICS (SMFSTAT)............................1,3,4,5,6
STATISTICS TIME INTERVAL IN MINUTES (STATIME).................1
SYNCHRONIZATION INTERVAL WITHIN THE HOUR (SYNCVAL)...........NO
ONLINE DATASET STATISTICS TIME INTERVAL IN MIN. (DSSTIME)......5
START MONITOR TRACE (MON).....................................1
MONITOR BUFFER SIZE IN BYTES (MONSIZE)................1,048,576
UNICODE IFCIDS (UIFCIDS)....................................YES
DDF/RRSAF ACCUM (ACCUMACC)...................................10
AGGREGATION FIELDS (ACCUMUID).................................0
COMPRESS SMF RECS (SMFCOMP)................................... ON
```

START AUDIT TRACE (AUDITST)

Shows whether the audit trace is started automatically when DB2 is started.

When YES, the audit trace is started for the default class (class 1) whenever DB2 is started. When ALL, an audit trace is automatically started for all classes.

Install parameter AUDIT TRACE on panel DSNTIPN, or ZPARM AUDITST in DSN6SYSP.

Field Name: QWP1AUDT

START GLOBAL TRACE (TRACSTR)

Shows whether the global trace is started automatically when DB2 is started.

When YES, the global trace starts for the default classes (classes 1, 2, and 3) whenever DB2 is started, and additional data consistency checks are made whenever a data page or index page is modified. When ALL, the global trace is automatically started for all classes.

The global trace is used to diagnose problems in DB2 but it also impacts DB2 performance. If you have production systems requiring high performance, you might consider turning off global trace. If you do this, be aware that this presents a serviceability exposure. In the event of a system failure, IBM service personnel will ask you to turn on global trace and attempt to recreate the problem.
Tracing, Checkpoint & Pseudo-Close Parameters (DSNTIPN)

Install parameter TRACE AUTO START on panel DSNTIPN, or ZPARM TRACSTR in DSN6SYSP.

**Field Name:** QWP1TRST

**TRACE TABLE SIZE IN 4K BYTES (TRACTBL)**

Shows the size of the RES trace table in 4 KB blocks. A value of 16 means 64 KB have been allocated for this table.

This is the default destination for the global trace records in DB2. Most trace records require 32-byte entries; events with more than three data items require 64-byte entries.

Install parameter TRACE SIZE on panel DSNTIPN, or ZPARM TRACTBL in DSN6SYSP.

**Field Name:** QWP1TRSZ

**START SMF ACCOUNTING (SMFACCT)**

Shows whether DB2 sends accounting data to SMF automatically when DB2 is started. Numeric values show what classes are sent. When YES, the default class (class 1) is sent. When ALL, accounting classes one through five are started.

The SMFPRM xx member of SYS1.PARMLIB must also be set to allow SMF to write the records.

Install parameter SMF ACCOUNTING on panel DSNTIPN, or ZPARM SMFACCT in DSN6SYSP.

**Field Name:** QWP1SMFA

**START SMF STATISTICS (SMFSTAT)**

Shows whether a Statistics trace was started automatically at DB2 startup time.

The classes started are shown separated by commas.

DB2 sends collected trace data to SMF. The SMFPRM xx member of SYS1.PARMLIB must be set to allow SMF to write the records.

Install parameter SMF STATISTICS on panel DSNTIPN, or ZPARM SMFSTAT in DSN6SYSP.

**Field Name:** QWP1SMFS

**STATISTICS TIME INTERVAL IN MINUTES (STATIME)**

The time interval, in minutes, between statistics collections. Statistics records are written approximately at the end of this interval.

Install parameter STATISTICS TIME on panel DSNTIPN, or ZPARM STATIME in DSN6SYSP.

**Field Name:** QWP1STIM

**SYNCHRONIZATION INTERVAL WITHIN THE HOUR (SYNCVAL)**

Shows whether DB2 statistics recording is synchronized with some part of the hour. The installation can specify that the DB2 statistics recording interval be synchronized with the beginning of the hour (00 minutes past the hour) or any number of minutes past the hour up to 59. Possible values are: 0-59, which indicate the synchronization point. When NO or N/A is shown, synchronization is disabled, this is the default.
Tracing, Checkpoint & Pseudo-Close Parameters (DSNTIPN)

If STATISTICS TIME INTERVAL IN MINUTES (STATIME) is greater than 60, NO or N/A is shown.

Install parameter STATISTICS SYNC on panel DSNTIPN, or ZPARM SYNCVAL in DSN6SYSP.

Field Name: QWP1SYNV

ONLINE DATASET STATISTICS TIME INTERVAL IN MIN. (DSSTIME)

The time interval, in minutes, before DB2 resets data set statistics collected for the online performance monitors. Online performance monitors can request DB2 data set statistics for the current interval with an IFI READS request for IFCID 199.

Install parameter DATASET STATS TIME on panel DSNTIPN, or ZPARM DSSTIME in DSN6SYSP.

Field Name: QWP1DTIM

START MONITOR TRACE (MON)

Shows whether the monitor trace is started automatically when DB2 is started. When YES, the default (trace class 1) is started. Numeric values show which classes are started. When ALL, monitor trace classes 1 through 8 are started.

Install parameter MONITOR TRACE on panel DSNTIPN, or ZPARM MON in DSN6SYSP.

Field Name: QWP1MON

MONITOR BUFFER SIZE IN BYTES (MONSIZE)

The default number of bytes allocated for the monitor trace buffer.

Install parameter MONITOR SIZE on panel DSNTIPN, or ZPARM MONSIZE in DSN6SYSP.

Field Name: QWP1MONS

UNICODE IFCIDS (UIFCIDS)

Shows whether output from IFC records should include Unicode information. Only a subset of the character fields (identified in the IFCID record definition by a %U in the comment area to the right of the field declaration in the DSNDQWxx copy files) are encoded in Unicode. The remaining fields maintain the same encoding of previous releases.

Install parameter UNICODE IFCIDS on panel DSNTIPN, or ZPARM UIFCIDS in DSN6SYSP.

Field Name: QWP1_UNICODE

DDF/RRSAF ACCUM (ACCUMACC)

Shows whether DB2 accounting data for DDF and RRSAF threads is accumulated by end user.

When NO, DB2 writes an accounting record when a DDF thread is made inactive, or when signon occurs for an RRSAF thread. A value in the range 2 through 65535 shows the number of times an end-user identifier should occur before DB2 writes an accounting record. An end-user identifier is the concatenation of the end-user user ID, end-user transaction name, and the end-user workstation name.
These values can be set by DDF threads using SERVER CONNECT and SET CLIENT calls, and by RRSAF threads using the RRSAF SIGN, AUTH SIGNON, and CONTEXT SIGNON functions.

An accounting record might be written prior to the number of end user occurrences in the following instances:

- When an internal storage threshold is reached for the accounting RRSAF signon call.
- When the thread deallocates, the accumulated accounting data for all end users on this thread is written (one record per end user).
- When this parameter is dynamically changed to deactivate accounting accumulation. In this instance, the next end-UR (for DDF thread) or signon (for a RRSAF thread) causes DB2 to write the accumulated accounting data for all end users on this thread (one record per end user).

Install parameter DDF/RRSAF ACCUM on installation panel DSNTIPN, or ZPARM ACCUMACC in DSN6SYSP.

**Field Name:** QWP1ACCU

**AGGREGATION FIELDS (ACCUMUID)**

Shows the aggregation fields used for DDF and RRSAF accounting rollup. Values are defined as follows:

- **0**  End user ID, transaction name, and workstation name
- **1**  End user ID
- **2**  End user transaction name
- **3**  End user workstation name
- **4**  End user ID and transaction name
- **5**  End user ID and workstation name
- **6**  End user transaction name and workstation name

This value is ignored if DDF or RRSAF accounting are not used. DB2 writes individual accounting threads for threads that do not have all aggregation fields populated that are specified by this parameter.

Install parameter AGGREGATION FIELDS on installation panel DSNTIPN, or ZPARM ACCUMUID in DSN6SYSP.

**Field Name:** QWP1ACID

**COMPRESS SMF RECS (SMFCOMP)**

Shows the COMPRESS DEST(SMF) TRACE records. This field corresponds to field COMPRESS SMF RECS on installation panel DSNTIPN. ZPARM name: SMFCOMP in DSN6SYSP.

**Field Name:** QWP1CSMF
Workfile Database Panel (DSNTIP91)

This topic shows detailed information about “System Parameters - Workfile Database Panel (DSNTIP91)”.

System Parameters - Workfile Database Panel (DSNTIP91)

The field labels shown in the following sample layout of “System Parameters - Workfile Database Panel (DSNTIP91)” are described in the following section.

WORKFILE DATABASE PANEL (DSNTIP91)
----------------------------------
MAX TEMP STORAGE PER AGENT IN MB (MAXTEMPS).................0
MAX TEMP RID (MAXTEMPS_RID).............................NOLIMIT
AGENT LEVEL THRESHOLD (WFSTGUSE_AGENT_THRESHOLD).............0
SYSTEM LEVEL THRESHOLD (WFSTGUSE_SYSTEM_THRESHOLD)..........90

MAX TEMP STORAGE PER AGENT IN MB (MAXTEMPS)

The maximum amount of temporary storage in megabytes (MB) for each agent.

Install parameter MAX TEMP STORAGE on panel DSNTIP6 or ZPARM MAXTEMPS in DSNTIP9.

Field Name: QWP4WFAL

MAX TEMP RID (MAXTEMPS_RID)

The maximum number of RID blocks of temporary storage in the Workfile database that a single RID list can use at any point in time. This field corresponds to field MAX TEMP RID on installation panel DSNTIP9. The ZPARM name is MAXTEMPS_RID.

It can have the following values:
• -1 if MAXTEMPS_RID=NONE
• 0 if MAXTEMPS_RID=NOLIMIT
• 1 to 329166 otherwise

Field Name: QWP4WFRD

AGENT LEVEL THRESHOLD (WFSTGUSE_AGENT_THRESHOLD)

Specifies the percentage of space that is used in the Workfile Database by a single agent when DB2 issues a warning message.

This value corresponds to field AGENT LEVEL THRESHOLD on installation panel DSNTIP9. The ZPARM name is WFSTGUSE_AGENT_THRESHOLD in DSN6SPRM.

Field Name: QWP4WFSAT

SYSTEM LEVEL THRESHOLD (WFSTGUSE_SYSTEM_THRESHOLD)

Specifies the percentage of space that is used in the Workfile Database by all agents in a DB2 subsystem or data sharing member when DB2 issues a warning message.

This value corresponds to field SYSTEM LEVEL THRESHOLD on installation panel DSNTIP9. The ZPARM name is WFSTGUSE_SYSTEM_THRESHOLD in DSN6SPRM.

Field Name: QWP4WFSST
Chapter 55. Alter Buffer Pool Command Issued

This topic shows detailed information about “System Parameters - Alter Buffer Pool Command Issued”.

System Parameters - Alter Buffer Pool Command Issued

The field labels shown in the following sample layout of “System Parameters - Alter Buffer Pool Command Issued” are described in the following section.

<table>
<thead>
<tr>
<th>Buffer Pool ID</th>
<th>Vpool Size (Pages) (Old)</th>
<th>Vpool Size (Pages) (New)</th>
</tr>
</thead>
<tbody>
<tr>
<td>02/12/13 15:03:23.56</td>
<td>BP25</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Field Name: QW0201BP</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Field Name: QW0201OP</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Field Name: QW0201NP</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Field Name: RT0201VS</th>
</tr>
</thead>
</table>

**BUFFER POOL ID**

The buffer pool internal identifier. The values 0 through 49 are the identifiers for BP0 through BP49. The values 80 through 89 are the identifiers for BF32K through BF32K9.

**VPOOL SIZE (PAGES) (OLD)**

The size of the old virtual pool.

**Field Name: QW0201OP**

**VPOOL SIZE (PAGES) (NEW)**

The size of the new virtual buffer pool.

**Field Name: QW0201NP**

**VPOOL SEQ THRESH**

The old and new virtual pool sequential steal threshold.

Old status taken from the DB2 field QW0201OT.

New status taken from the DB2 field QW0201NT.

**Field Name: RT0201VS**

**HORIZ DEFER WRITE THRESH**

This threshold is a percentage of the virtual buffer pool that might be occupied by unavailable pages, including updated pages and pages in use.

The default value for QWQT is 50%. You can change this value to any value from 0% to 90% using the DWQT option of the ALTER BUFFERPOOL command.

DB2 checks QWQT when an update to a page is complete. If the percentage of unavailable pages in the virtual buffer pool exceeds QWQT, write operations are scheduled for up to 128 pages per data set to decrease
Alter Buffer Pool Command Issued

the number of unavailable buffers to 10% below QWQT. For example, if QWQT is 50%, the number of unavailable buffers is reduced to 40%.

When the limit of QWQT is reached, data sets containing the oldest updated pages are written asynchronously. DB2 continues to write pages until the ratio goes below the QWQT.

Field Name: QDBPDWQT

VERT DEFER WRITE THRESH (%)

The vertical deferred write threshold for the virtual buffer pool expressed as percentage.

Old status taken from the DB2 field QW0201OV.
New status taken from the DB2 field QW0201NV.

Field Name: RT0201PC

VERT DEFER WRITE THRESH (BUF)

The vertical deferred write threshold for the virtual buffer pool expressed as an absolute number of buffers. It is only used if VERTICAL DEFERRED WRITE THRESHOLD (PERCENTAGE) is 0.

Old status taken from the DB2 field QW0201OJ.
New status taken from the DB2 field QW0201NJ.

Field Name: RT0201BU

VPOOL PARALLEL SEQ THRESH

The old and new virtual pool parallel sequential threshold.

Old status taken from the DB2 field QW0201OQ.
New status taken from the DB2 field QW0201NQ.

Field Name: RT0201VP

ASSISTING PARALLEL SEQ THRESH

The assisting parallel sequential threshold before and after the ALTER BUFFERPOOL command was issued.

Old status taken from the DB2 field QW0201OX.
New status taken from the DB2 field QW0201NX.

Field Name: RT0201AS

PGFIX ATTRIBUTE

Indicates whether a page is fixed in real storage when it is first used. It can have one of the following values: YES or NO.

Field Name: QDBPFPFIX

PAGE STEAL METHOD

Identifies the page stealing algorithm (PGSTEAL) that is used for the virtual buffer pool. It controls when and whether performance-critical objects in buffer pools are removed from buffer pools when the space is needed by other objects. Possible values are:

LRU Least recently used (LRU) objects are removed first. This means it takes away pages that are not used so that more recently used pages can remain in the virtual buffer pool. This is used by default.
FIFO  
First-In-First-Out (FIFO) means that the oldest objects are removed first. This results in a small decrease in the cost of a Getpage operation. It can reduce internal DB2 latch contention in environments that require very high concurrency.

NONE
Objects are not removed from buffer pool (no page stealing). This setting provides the highest availability for business-critical objects.

Old status taken from the DB2 field QW0201OK.
New status taken from the DB2 field QW0201NK.
Field Name: RT0201PS

AUTOSIZE
The old and new status of the AUTOSIZE attribute.
Old status taken from the DB2 field QW0201OZ.
New status taken from the DB2 field QW0201NZ.
Field Name: RT0201AT

FRAMESIZE (OLD)
The size of the old frame (4 KB, 1 MB, or 2 GB).
Field Name: QW0201OC

FRAMESIZE (NEW)
The new frame size (4 KB, 1MB, or 2GB).
Field Name: QW0201NC

VPOOL SIZE MIN (OLD)
The minimum size of the old virtual pool.
Field Name: QW0201OA

VPOOL SIZE MIN (NEW)
The minimum size of the new virtual pool.
Field Name: QW0201NA

VPOOL SIZE MAX (OLD)
The maximum size of the old virtual pool.
Field Name: QW0201OB

VPOOL SIZE MAX (NEW)
The maximum size of the new virtual pool.
Field Name: QW0201NB
Accounting Report
Chapter 56. Alter Group Buffer Pool Command Issued

This topic shows detailed information about “System Parameters - Alter Group Buffer Pool Command Issued”.

System Parameters - Alter Group Buffer Pool Command Issued

The field labels shown in the following sample layout of “System Parameters - Alter Group Buffer Pool Command Issued” are described in the following section.

<table>
<thead>
<tr>
<th>ALTER GROUP BUFFER POOL COMMAND ISSUED</th>
<th>OLD</th>
<th>NEW</th>
</tr>
</thead>
<tbody>
<tr>
<td>TIMESTAMP</td>
<td>02/15/13 13:52:59.07</td>
<td></td>
</tr>
<tr>
<td>MEMBER</td>
<td>SE11</td>
<td></td>
</tr>
<tr>
<td>GBP ID</td>
<td>GBP0</td>
<td></td>
</tr>
<tr>
<td>CURRENT DIRECTORY TO DATA RATIO</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>CLASS CASTOUT THRESHOLD (%)</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>CLASS CASTOUT THRESHOLD (PAGES)</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>GBP CASTOUT THRESHOLD (%)</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>GBP CHECKPOINT INTERVAL (MIN)</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>GBP CACHE SETTING</td>
<td>YES</td>
<td>YES</td>
</tr>
<tr>
<td>AUTO REC</td>
<td>YES</td>
<td>YES</td>
</tr>
</tbody>
</table>

GBP ID

The DB2 group buffer pool ID.

Field Name: QW0256GB

CURRENT DIRECTORY TO DATA RATIO

The directory entry to data entry ratio. This is the value specified in the RATIO keyword of the ALTER GROUPBUFFERPOOL command.

New status deduced from the DB2 field QW0256NR.
Old status deduced from the DB2 field QW0256OR.

Field Name: RT0256DR

CLASS CASTOUT THRESHOLD (%)

The threshold at which the class castout is to be initiated. It is expressed as a percentage of the group buffer pool size. This is the value specified in the CLASST keyword of the ALTER GROUPBUFFERPOOL command.

New status deduced from the DB2 field QW0256NC.
Old status deduced from the DB2 field QW0256OC.

Field Name: RT0256CT

CLASS CASTOUT THRESHOLD (PAGES) (OLD)

The old class castout threshold based on the number of pages.

Field Name: QW0256ON

CLASS CASTOUT THRESHOLD (PAGES) (NEW)

The new class castout threshold based on the number of pages.

Field Name: QW0256NN

GBP CASTOUT THRESHOLD (%)

The threshold at which the castout is to be initiated for the group buffer pool. This is the value specified in the GBPOOLT keyword of the ALTER GROUPBUFFERPOOL command.

New status deduced from the DB2 field QW0256NG.
Alter Group Buffer Pool Command Issued

Old status deduced from the DB2 field QW0256OG.

Field Name: RT0256GT

GBP CHECKPOINT INTERVAL (MIN)

The time interval (in minutes) between successive group buffer pool checkpoints. This is the value specified in the GBPCHKPT keyword of the ALTER GROUPBUFFERPOOL command.

New status deduced from the DB2 field QW0256NK.

Old status deduced from the DB2 field QW0256OK.

Field Name: RT0256CI

GBP CACHE SETTING

GBPCACHE value before and after the ALTER GROUPBUFFERPOOL command was issued. This field specifies whether DB2 should write changed pages for the group buffer pool dependant pageset or partitions directly to DASD and use the group buffer pool only for sending XI signals.

New status deduced from the DB2 field QW0256NB.

Old status deduced from the DB2 field QW0256OB.

Field Name: RT0256CS

AUTO REC

A flag indicating how the AUTOREC option of the ALTER GROUPBUFFERPOOL command has been set. It specifies whether DB2 should automatically recover if GBP fails. The old value specifies the AUTOREC value before the ALTER GBP command was issued. The new value specifies the AUTOREC value after the ALTER GBP command was issued.

New status deduced from the DB2 field QW0256NA.

Old status deduced from the DB2 field QW0256OA.

Field Name: RT0256AR
Chapter 57. Buffer Pool Parameters

This topic shows detailed information about “System Parameters - Buffer Pool Parameters”.

Normally, buffer pool information is reported once for each buffer pool if the attributes remain the same over the reporting period.

Buffer pool attributes can be changed while DB2 is active using the DB2 ALTER BUFFERPOOL command. If the performance trace class 10 is active, the event is recorded in the system parameters report.

When an ALTER BUFFERPOOL command is recorded, the status of the buffer pool before and after the command is shown.

**Note:** The fields shown on this panel depend on the installed DB2 version.

**System Parameters - Buffer Pool Parameters**

The field labels shown in the following sample layout of “System Parameters - Buffer Pool Parameters” are described in the following section.

<table>
<thead>
<tr>
<th>BUFFER POOL PARAMETERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>TIMESTAMP</td>
</tr>
<tr>
<td>BUFFER POOL ID</td>
</tr>
<tr>
<td>VPOOL SIZE (PAGES)</td>
</tr>
<tr>
<td>VPOOL SEQ THRESH</td>
</tr>
<tr>
<td>HORIZ DEFER WRITE THRESH</td>
</tr>
<tr>
<td>VERT DEFER WRITE THRESH (%)</td>
</tr>
<tr>
<td>VERT DEFER WRITE THRESH (BUF)</td>
</tr>
<tr>
<td>VPOOL PARALLEL SEQ THRESH</td>
</tr>
<tr>
<td>ASSISTING PARALLEL SEQ THRESH</td>
</tr>
<tr>
<td>PGFIX ATTRIBUTE</td>
</tr>
<tr>
<td>PAGE STEAL METHOD</td>
</tr>
<tr>
<td>AUTOSIZE</td>
</tr>
<tr>
<td>FRAMESIZE</td>
</tr>
<tr>
<td>VPOOL SIZE MIN</td>
</tr>
<tr>
<td>VPOOL SIZE MAX</td>
</tr>
</tbody>
</table>

**BUFFER POOL ID**

Buffer pool name.

**Field Name:** QDBPNM

**VPOOL SIZE (PAGES)**

The size of the virtual buffer pool.

Old status taken from the DB2 field QW0201OP.

New status taken from the DB2 field QW0201NP.

**Field Name:** QDBPVPVSZ

**VPOOL SEQ THRESH**

Virtual pool sequential threshold (VPSEQT). This threshold is a percentage of the virtual buffer pool that might be occupied by sequentially accessed pages. The pages can be in the state updated, in use, or available. Therefore, each page might count regarding exceeding any other buffer pool threshold.

The default value for VPSEQT is 80%. You can change this value to a value from 0% to 100% by using the VPSEQT option of the ALTER BUFFERPOOL command.
Buffer Pool Parameters

VPSEQT is checked before stealing a buffer for a sequentially accessed page instead of accessing the page in the virtual buffer pool. If the threshold is exceeded, DB2 tries to steal a buffer that holds a sequentially accessed page rather than one that holds a randomly accessed page.

If you set VPSEQT to 0%, sequential pages cannot occupy space in the virtual buffer pool. In this case, prefetch is disabled, and sequentially accessed pages are discarded when they are released. If you set VPSEQT to 0%, the value of HPSEQT is meaningless because sequential pages that are not kept in the virtual buffer pool do not go in the hiperpool. You can, however, set the value for HPSEQT to a value above zero and the value for VPSEQT to zero. If you set VPSEQT to 100%, sequential pages can monopolize the entire virtual buffer pool.

Field Name: QDBPVPSH

HORIZ DEFER WRITE THRESH

This threshold is a percentage of the virtual buffer pool that might be occupied by unavailable pages, including updated pages and pages in use.

The default value for QWQT is 50%. You can change this value to any value from 0% to 90% using the DWQT option of the ALTER BUFFERPOOL command.

DB2 checks QWQT when an update to a page is complete. If the percentage of unavailable pages in the virtual buffer pool exceeds QWQT, write operations are scheduled for up to 128 pages per data set to decrease the number of unavailable buffers to 10% below QWQT. For example, if QWQT is 50%, the number of unavailable buffers is reduced to 40%.

When the limit of QWQT is reached, data sets containing the oldest updated pages are written asynchronously. DB2 continues to write pages until the ratio goes below the QWQT.

Field Name: QDBPDWQT

VERT DEFER WRITE THRESH (%)

Vertical deferred write threshold (VDWQT). This threshold is similar to the deferred write threshold but it applies to the number of updated pages for one single page set in the buffer pool. If the percentage or number of updated pages for the data set exceeds the threshold, writes up to 128 pages are scheduled for that data set.

VDWQT can be specified in one of the following ways:

- As a percentage of the virtual buffer pool that might be occupied by updated pages from one single page set. The default value for this threshold is 10%. You can change the percentage to any value from 0% to 90%.
- As the total number of buffers in the virtual buffer pool that might be occupied by updated pages from one single page set. You can specify the number of buffers from 0 to 9999. If you want to use the number of buffers as your threshold, you must set the percentage threshold to 0.

Field Name: QDBPVDQT

VERT DEFER WRITE THRESH (BUF)

The vertical deferred write threshold (VDWQT), shown as the number of buffers in the virtual buffer pool that might be occupied by updated pages from a single page set.
Buffer Pool Parameters

Field Name: QDBPVDQB

VPOOL PARALLEL SEQ THRESH

Virtual buffer pool parallel sequential threshold (VPSEQT). This threshold is a part of the virtual buffer pool that might support parallel operations. It is measured as a percentage of the sequential steal threshold (VPSEQT). Setting VPSEQT to zero disables parallel operation.

The default value for this threshold is 50% of the sequential steal threshold (VPSEQT). You can change the default value to any value from 0% to 100% by using the VPSEQT option on the ALTER BUFFERPOOL command.

Field Name: QDBPPSQT

ASSISTING PARALLEL SEQ THRESH

Virtual buffer pool assisting parallel sequential threshold (VXPSEQT). This threshold is a part of the virtual buffer pool that might support parallel operations initiated from another DB2 in the data sharing group. It is measured as a percentage of VPPSEQT.

Setting VXPSEQT to zero (default) prevents DB2 from supporting sysplex query parallelism at run time for queries that use this buffer pool.

You can change the default value to any value from 0% to 100% using the VXPSEQT option of the ALTER BUFFERPOOL command.

Field Name: QDBPXSQT

PGFIX ATTRIBUTE

Indicates whether a page is fixed in real storage when it is first used. It can have one of the following values: YES or NO.

Field Name: QDBPFPFIX

PAGE STEAL METHOD

Identifies the page stealing algorithm (PGSTEAL) that is used for the virtual buffer pool. It controls when and whether performance-critical objects in buffer pools are removed from buffer pools when the space is needed by other objects. Possible values are:

- **LRU** Least recently used (LRU) objects are removed first. This means it takes away pages that are not used so that more recently used pages can remain in the virtual buffer pool. This is used by default.

- **FIFO** First-In-First-Out (FIFO) means that the oldest objects are removed first. This results in a small decrease in the cost of a Getpage operation. It can reduce internal DB2 latch contention in environments that require very high concurrency.

- **NONE** Objects are not removed from buffer pool (no page stealing). This setting provides the highest availability for business-critical objects.

Old status taken from the DB2 field QW0201OK.
New status taken from the DB2 field QW0201NK.

Field Name: RT0201PS

AUTOSIZE

Indicates if the AUTOSIZE option is activated on the ALTER BUFFERPOOL command.
Buffer Pool Parameters

Field Name: QDBPASIZ
FRAMESIZE
  The frame size.
Field Name: QDBPFRAM
VPOOL SIZE MIN
  The minimum size of the virtual pool.
Field Name: QDBPVPMI
VPOOL SIZE MAX
  The maximum size of the virtual pool.
Field Name: QDBPVPMX
Chapter 58. Group Buffer Pool Parameters

This topic shows detailed information about “System Parameters - Group Buffer Pool Parameters”.

This block shows the merged group buffer pool data from all the members of a DB2 data sharing group. To produce this report, statistics class 5 must be active.

Each time an ALTER GROUPBUFFERPOOL command is issued for a member, an IFCID 230 record is produced showing information about the group buffer pools connected to that particular member of a data sharing group. If the IFCID 230 record indicates that the status of the group buffer pools has changed since the last IFCID 230 record was produced (regardless of which member produced it), or if this is the first IFCID 230 encountered, the new status of the group buffer pools is printed.

The status of the group buffer pools changes if the IFCID 230 record indicates one of the following:
- A member uses a new group buffer pool.
- A member does not use a group buffer pool that it used previously.
- At least one of the group buffer pool attributes has changed.

All the group buffer pools connected to the member, whether or not they have changed, are printed.

System Parameters - Group Buffer Pool Parameters

The field labels shown in the following sample layout of “System Parameters - Group Buffer Pool Parameters” are described in the following section.

CURRENT DIRECTORY TO DATA RATIO

The current directory entry to data entry ratio.

For ALTER GROUPBUFFERPOOL commands, this field reports the value specified in the RATIO keyword.
Group Buffer Pool Parameters

Field Name: QBGBGR1

CLASS CASTOUT THRESHOLD (%)
The threshold at which the class castout is to be initiated. It is expressed as a percentage of the size of the group buffer pool.

For ALTER GROUPBUFFERPOOL commands, it reports the value specified in the CLASST keyword.

Field Name: QBGBGCT

GBP ID
Group buffer pool name.

Field Name: QBGBGN

DB2 11: CLASS CASTOUT THRESHOLD (PAGES)
The class castout threshold based on the number of pages.

Field Name: QBGBGCTN

ALLOCATED GBP SIZE (4K)
The allocated size of the group buffer pool in 4 KB blocks.

Field Name: QBGBGSZ

GBP CASTOUT THRESHOLD (%)
The threshold at which the castout is to be initiated for the group buffer pool. It is expressed as a percentage of the size of the group buffer pool.

For ALTER GROUPBUFFERPOOL commands, it reports the value specified in the GBPOOLT keyword.

Field Name: QBGBGGT

ACTUAL DIRECTORY
The actual number of allocated directory entries.

Field Name: QBGBGDR

GBP CHECKPOINT INTERVAL (MIN)
The time interval, in minutes, between successive group buffer pool checkpoints.

For ALTER GROUPBUFFERPOOL commands, it reports the value specified in the GBPCHKPT keyword.

Field Name: QBGBGCK

ACTUAL DATA ENTRY
The actual number of allocated data entries.

Field Name: QBGBGDT

GBP CACHE SETTING
GBP cache attribute. Possible values are:

YES GBP is used for both data caching and cross-invalidation.
NO GBP is used for cross-invalidation only.

Field Name: QBGBGCS
Group Buffer Pool Parameters

PENDING DIRECTORY TO DATA RATIO
The pending directory entry to data entry ratio.
Field Name: QBGBGR2

AUTO REC
Indicates whether automatic recovery takes place in the event of a structure failure or a loss of connectivity. When automatic recovery is active, all members of the group are recovered to the group buffer pool.
Field Name: QBGBGAS

MODE
Simplex or duplex mode indicator.
Field Name: QBGBDUP

Prior to DB2 11: SEC-GBP ALLOC
The allocated size of the secondary GBP when the GBP is DUBLEX.
This field is not shown when MODE is SIMPLEX.
Field Name: QBGBGSZ2

Prior to DB2 11: SEC-GBP ALLOC DIRECTORY ENTRY
Number of allocated directory entries in the secondary GBP when MODE is DUBLEX.
Field Name: QBGBGDR2

Prior to DB2 11: SEC-GBP ALLOC DATA ENTRY
The allocated data entries in the secondary GBP when MODE is DUBLEX.
Field Name: QBGBGDT2
Part 11. The Utility Activity Report Set

These topics provide information about the Utility Activity reports.

Note: For an introduction to the Utility Activity report set and general Utility Activity information refer to the Reporting User’s Guide. It also provides information on input to Utility Activity reports.

Chapter 59, “Headers Used in Utility Activity,” on page 59-1
OMEGAMON XE for DB2 PE header information is printed at the top of each Utility Activity report or trace page.

Chapter 60, “The Utility Activity Reports,” on page 60-1
This topic describes the Utility Activity report.

Chapter 61, “The Utility Activity Trace,” on page 61-1
This topic introduces the Utility Activity trace.

Chapter 62, “Workload Detail,” on page 62-1
Workload detail blocks are only printed in Utility Activity traces.
Chapter 59. Headers Used in Utility Activity

OMEGAMON XE for DB2 PE header information is printed at the top of each Utility Activity report or trace page.

There are two types of headers:
- The Utility Activity report header
- The Utility Activity trace header.

The report and trace header shows the following information:

LOCATION
The DB2 reporting location. If the location name is not available, the DB2 data sharing group name is printed in this field. If the DB2 data sharing group name does not exist, the DB2 subsystem ID is printed.

OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (VnRnMn)
The product name and the version, release, and modification level.

PAGE
The page number in the format lii-nnnnnn, where lii denotes the location number within the report and nnnnnn the page number within the location.

GROUP
The name of the DB2 data sharing group. This field shows N/A if there is no group name.

REQUESTED FROM and TO
The FROM and TO dates and times specified in the REPORT or TRACE subcommand.

If both FROM and TO dates and times are omitted from the REPORT subcommand, the FROM and TO dates and times specified in GLOBAL are printed. If only the FROM date and time or only the TO date and time has been specified, NOT SPECIFIED is printed for the unspecified value.

If FROM and TO are not specified in REPORT or GLOBAL, NOT SPECIFIED appears for both the FROM and TO values.

If you have specified FROM and TO times without dates in REPORT or GLOBAL, ALL DATES is printed along with the specified times.

MEMBER
The name of the DB2 data sharing member or the member name of the DB2 subsystem. This field shows N/A if there is no member name.

This field is not printed on the report page showing the GROUP BUFFERPOOLS PARAMETERS.

SUBSYSTEM
The ID of the DB2 subsystem that generated the data. This field is not printed on the report page showing the GROUP BUFFERPOOLS PARAMETERS.

ORDER
If the ORDER option of the REPORT or TRACE subcommand was used to arrange the report entries, the selected keywords are shown in this field. Depending on the context, the OMEGAMON XE for DB2 PE identifiers by which lock events are grouped are shown here.
Utility Activity - Report Headers

ACTUAL FROM/TO
The date and time of the first and last record included in the log for a location, group, subsystem, or member.

DB2 VERSION
The DB2 version number of the subsystem that generated the data.

PAGE DATE
The date of the timestamps printed on this page. A page break occurs at the change of the date. This is useful if a trace page contains more than one entry and the date is not shown for each entry.

IDENTIFIED BY
Shows the identifiers specified with the ORDER option.

WITH detail WORKLOAD
The workload details as specified on the WORKLOAD option of the TRACE subcommand.

"Utility Activity Report Header Example" on page 59-3
The Utility Activity reports contain information in the header at the top of each page as shown in the following example.

"Utility Activity Trace Header" on page 59-4
This section introduces the header of the Utility Activity trace.
Utility Activity Report Header Example

The Utility Activity reports contain information in the header at the top of each page as shown in the following example.

LOCATION: USIBMSYSTD02
GROUP: OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0)
SUBSYSTEM: SSDQ
DB2 VERSION: V10

GROUP: DSNCAT
MEMBER: SSDQ
ORDER: PRIMAUTH-PLANNAME

PAGE: 1-1
REQUESTED FROM: NOT SPECIFIED
TO: NOT SPECIFIED
ACTUAL FROM: 01/30/10 23:50:43.70
TO: 01/30/10 02:35:57.68
Utility Activity Trace Header

This section introduces the header of the Utility Activity trace.

Utility Activity Trace Header Example

The Utility Activity trace header contains the following information.
Chapter 60. The Utility Activity Reports

This topic describes the Utility Activity report.

The following command generates the Utility Activity report shown in "Utility Activity Report Example":

```
UTILITY
REPORT
```

This command produces a report including both BIND and UTILITY activity types by default.

**Utility Activity Report Example**

<table>
<thead>
<tr>
<th>LOCATION: PMODA22</th>
<th>OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0)</th>
<th>PAGE: 1-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>GROUP: N/P</td>
<td>UTILITY ACTIVITY REPORT</td>
<td>REQUESTED</td>
</tr>
<tr>
<td>SUBSYSTEM: DA22</td>
<td>ORDER: PRIMAUTH-PLANNAME</td>
<td>TO: NOT SPECIFIED</td>
</tr>
<tr>
<td>DB2 VERSION: V10</td>
<td>ACTUAL FROM: 05/31/13 09:01:18.84</td>
<td>ACTUAL FROM: 05/31/13 09:01:18.84</td>
</tr>
<tr>
<td></td>
<td>TO: 01/02/00 00:00:00.00</td>
<td>TO: 01/02/00 00:00:00.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PRIMAUTH</th>
<th>PLANNAME</th>
<th>OCCURRENCES</th>
<th>TOT. ELAPSED</th>
<th>TOT. CP CPU</th>
<th>TOT. ZIIP</th>
<th>AVG. ELAPSED</th>
<th>AVG. CP CPU</th>
<th>AVG. ZIIP</th>
<th>ACTIVITY TYPE</th>
<th>COUNT</th>
<th>AVG. ELAPSED</th>
<th>AVG. CP CPU</th>
<th>AVG. ZIIP</th>
<th>ACTIVITY TYPE</th>
<th>COUNT</th>
<th>AVG. ELAPSED</th>
<th>AVG. CP CPU</th>
<th>AVG. ZIIP</th>
</tr>
</thead>
<tbody>
<tr>
<td>BBE</td>
<td>DSNUTIL</td>
<td>5</td>
<td>1.605797</td>
<td>0.104854</td>
<td>UTILITY</td>
<td>0.321159</td>
<td>0.038971</td>
<td>0.011058</td>
<td>0.000000</td>
<td>0.000000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>REORG</td>
<td>3</td>
<td>0.699737</td>
<td>0.113876</td>
<td>LOAD</td>
<td>2</td>
<td>0.906860</td>
<td>0.000978</td>
<td>0.001934</td>
<td>0.040489</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>LOAD</td>
<td>2</td>
<td>0.906860</td>
<td>0.000978</td>
<td>0.001934</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Field description**

The report contains the following fields:

**OMEGAMON XE for DB2 PE identifiers**

Up to three OMEGAMON XE for DB2 PE identifiers can be printed in this column. They are printed whenever they change. The second and third identifiers are indented. If the ORDER option is not used, the default of PLANNAME within PRIMAUTH is shown. Refer to Report Command Reference for more information about the ORDER option.

**OCCURRENCES**

The total number of bind or utility threads for the current combination of OMEGAMON XE for DB2 PE identifiers. A bind thread is identified by the presence of appropriate pairs of IFCIDs 108, 110, 111, 177, and 183. A utility thread is identified by the presence of IFCIDs 023, 024, and 025.

**TOT. ELAPSED**

The time difference between the first bind or utility record and the last bind or utility record.

**AVG. ELAPSED**

The TOT. ELAPSED time divided by OCCURRENCES.

**TOT. CP CPU**

The difference between the CPU time of the first bind or utility record and the CPU time of the last bind or utility record.
Utility Activity - Report

<table>
<thead>
<tr>
<th></th>
<th>AVG. CP CPU</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The TOT. CPUTIME divided by OCCURRENCES.</td>
</tr>
</tbody>
</table>

**ACTIVITY TYPE**

The name of the activity type and event. The activity type can only be BIND for bind events including remote bind activity, or UTILITY for utility events. All events are indented.

The bind events are as follows:

**BIND PLAN**
BIND PLAN subcommand issued

**BIND PACK**
BIND PACKAGE subcommand issued

**BIND R-PACK**
BIND PACKAGE subcommand issued for a remote location

**RBND PLAN**
REBIND PLAN subcommand issued

**RBND PACK**
REBIND PACKAGE subcommand issued

**RBND R-PACK**
REBIND PACKAGE subcommand issued for a remote location

**FREE PLAN**
FREE PLAN subcommand issued

**FREE PACK**
FREE PACKAGE subcommand issued

**FREE R-PACK**
FREE PACKAGE subcommand issued for a remote location

**CONNECT**
BIND CONNECT or CONNECT RESET subcommand issued for a remote location.

The utility events are as follows:

**CHECKDAT**
Identifies the utility as CHECK DATA.

**CHECKIDX**
Identifies the utility as CHECK INDEX.

**COPY**
Identifies the utility as COPY.

**DIAGNOSE**
Identifies the utility as DIAGNOSE.

**LOAD**
Identifies the utility as LOAD.

**MERGECOP**
Identifies the utility as MERGECOPY.

**MODIFY**
Identifies the utility as MODIFY.

**QUIESCE**
Identifies the utility as QUIESCE.
Utility Activity - Report

RECOVER
  Identifies the utility as RECOVER TABLESPACE.

RECOVERI
  Identifies the utility as RECOVER INDEX.

REBUILDI
  Identifies the utility as REBUILD INDEX.

REORG
  Identifies the utility as REORG.

REPAIR
  Identifies the utility as REPAIR.

REPORT
  Identifies the utility as REPORT.

RUNSTATS
  Identifies the utility as RUNSTATS.

STOSPACE
  Identifies the utility as STOSPACE.

UNLOAD
  Identifies the utility as UNLOAD.

COUNT
  The number of occurrences of a single bind or utility event.

TOT. ELAPSED
  The time difference between the first and last occurrence of a specific bind or utility event.

AVG. ELAPSED
  The TOT. ELAPSED time divided by COUNT.

TOT. CP CPU
  The difference between the CPU time of the first occurrence of a specific bind or utility event and the CPU time of the UTILEND of the last occurrence of this bind or utility event.

AVG. CP CPU
  The TOT. CP CPU time divided by COUNT.

TOT. SORTCPU
  The sum of the Sort CPU time for all occurrences of a specific utility event.

TOT. ZIIP
  The sum of the total utility ZIIP time for all occurrences of a specific utility event.

AVG. ZIIP
  The TOT. ZIIP time divided by COUNT.

SORT ZIIP
  The sum of the Sort ZIIP time for all occurrences of a specific utility event.
Chapter 61. The Utility Activity Trace

This topic introduces the Utility Activity trace.

Traces are presented in the order in which the threads complete. Start times might not be shown in ascending order if other threads finished prior to completion of a thread which started earlier.

The Utility Activity trace is generated with the following command:

```
UTILITY
TRACE
ORDER (PRIMAUTH-PLANNAME-INSTANCE)
```

This command produces a trace including both BIND and UTILITY activity types but excluding any workload detail by default.

Utility Activity Short Trace Example

Here is an example of a Utility Activity Short trace.

```
LOCATION: OMPDB51 OMEGAMON XE FOR DB2 PERFORMANCE EXPERT (V5R2M0) PAGE: 1-1
GROUP: N/P UTILITY ACTIVITY TRACE REQUESTED FROM: NOT SPECIFIED
MEMBER: N/P TO: NOT SPECIFIED
SUBSYSTEM: DB51 ACTUAL FROM: 06/07/13 09:14:49.15
DB2 VERSION: V11 PAGE DATE: 06/07/13
WORKLOAD(NONE)
IDENTIFIED BY PRIMAUTH/PLANNAME/INSTANCE
PRIMAUTH
PLANNAME ELAPSED TIME
INSTANCE START TIME CPU TIME ACTIVITY TYPE OBJECT(S)
---------------------- ----------- ------------ ----------------------- -----------------------------------------------------------
KOZS
DSNBIND
X'CB79D266C962' 09:15:02.92 0.000280 BIND
0.000000 BIND PLAN PLANNAME: DSNREXX
KOZS
DSNBIND
X'CB79D3411068' 09:18:51.94 0.027328 BIND
0.000000 RBND PACK LOCN: OMPDB51
COLL: UTRUN
PKID: UTILV110
VRID: N/P
CONS: X'195D036002523DD6'
```

Field description

The trace contains the following fields:
Utility Activity - Trace

**OMEGAMON XE for DB2 PE identifiers**
The OMEGAMON XE for DB2 PE identifiers specified in the ORDER option. They are printed whenever they change. The second and third identifiers are indented.

**PLANNAME**
The DB2 application plan name of the thread.

**ELAPSED TIME**
The time difference between START TIME and the timestamp of the ENDUTIL of the last bind or utility record of the originating task.

**JOB NAME**
User-defined job name.

**SHRLEVEL**
The SHRLEVEL value of the utility. Possible values are: NONE, REFERENCE, or CHANGE.

**START TIME**
The timestamp of the first bind or utility record encountered for that thread.

**CPU TIME**
The CPU time of the bind or utility event, including the CPU time of any parallel tasks.

**ACTIVITY TYPE**
The name of the activity type and event. The activity type can only be BIND for bind events including remote bind activity, or UTILITY for utility events. All events are indented.

For a detailed description of Activity Type, see page "Field description" on page 60-1.

**OBJECT(S)**
The description depends on whether it is a utility event, a bind event referring to a plan, a bind event referring to a package, or a BIND CONNECT:

- In the case of a utility, it is the database name.object name for each object worked on by the utility. Each database name.object name is shown only once.
- In the case of a BIND PLAN, RBND PLAN, or FREE PLAN event, it is the plan name.
- In the case of a BIND PACK, RBND PACK, FREE PACK, BIND R-PACK, RBND R-PACK, or FREE R-PACK event, the following information is shown:
  - **LOCN** The location of the package.
  - **COLL** The collection to which the package belongs.
  - **PKID** The package ID.
  - **VRID** The first 53 characters of the version name.
  - **CONS** The consistency token for the package.
- In the case of a CONNECT, it is LOCN, the location of the package.

**TOT. ZIIP**
The total utility ZIIP time (if Accounting class 1 trace is activated).
SORT  The following fields provide information about sorting. They are only written for utility events:

DF    Shows if DFSORT was invoked at least once (Y/N).

DB2   Shows if DB2SORT was invoked at least once (Y/N).

DATA  Shows the number of parallel data sorts.

INDEX Shows the number of parallel index sorts.

OTHER Shows the number of other sorts.

SORT CPU Shows the SORT CPU time.

SORT ZIIP Shows the SORT ZIIP time (if provided by the SORT program).
Accounting Report
Chapter 62. Workload Detail

Workload detail blocks are only printed in Utility Activity traces.

The workload detail blocks are displayed beneath the thread events.

"Bind Activity" on page 62-2
This topic describes the layout of the Bind Activity block.
"Data Set Information" on page 62-7
This block shows the data set information available for the activity.
"Exit Activity" on page 62-8
This block shows the exits performed by the event.
"I/O Activity" on page 62-9
This block shows the I/O activity for each object performed by the event.
"LISTDEF Information" on page 62-11
This topic describes the workload block of the list definition (LISTDEF) information.
"LockSuspension Activity" on page 62-12
This topic shows detailed information about “Workload Detail - Lock Suspension Activity”.
"Page and Row Locking Activity“ on page 62-14
This block shows the page locking, row locking, and lock avoidance activity for each object, performed by the event.
"Utility Phases” on page 62-17
This block shows the utility phases for each object performed by the event. Its layout depends on whether the utility produces parallel tasks.
Utility Activity - Bind Activity

Bind Activity

This topic describes the layout of the Bind Activity block.

This block shows the bind activity for:
- BIND PACK
- BIND R-PACK
- RBND PACK
- RBND R-PACK
- BIND PLAN
- RBND PLAN

The layout depends on whether it is a package or a plan for which bind activity is shown.

Bind Activity Workload Block Example for Packages or Plans

Here is an example of the Bind Activity Workload Block for Packages.

--- BIND ACTIVITY

<table>
<thead>
<tr>
<th>ISOLATION</th>
<th>CS</th>
<th>TYPE</th>
<th>AUTOMATIC</th>
<th>ACQUIRE</th>
<th>ALLOCATION</th>
<th>DEGREE</th>
<th>ANY</th>
<th>VALIDATE</th>
<th>BIND</th>
</tr>
</thead>
<tbody>
<tr>
<td>OWNER</td>
<td>MANFRED</td>
<td>CURRENTDATA</td>
<td>YES</td>
<td>RELEASE</td>
<td>DEALLOCATION</td>
<td>KEEPDYNAMIC</td>
<td>YES</td>
<td>EXPLAIN</td>
<td>YES</td>
</tr>
<tr>
<td>DYNAMICRULES</td>
<td>BIND</td>
<td>DISCONNECT</td>
<td>CONDITIONAL</td>
<td>PREPARE</td>
<td>NODEFER</td>
<td>QUALIFIER</td>
<td>HUGO</td>
<td>REOPTIMIZE</td>
<td>YES</td>
</tr>
<tr>
<td>ACTION</td>
<td>REPLACE</td>
<td>SQLERROR</td>
<td>NOPACKAGE</td>
<td>SQLRULES</td>
<td>DB2</td>
<td>PROTOCOL</td>
<td>NOT_SPEC</td>
<td>OPTHINT</td>
<td>YES</td>
</tr>
<tr>
<td>IMMEDIATE</td>
<td>YES</td>
<td>SENT</td>
<td>123456</td>
<td>ELAPSED_TIME: 1234.123456</td>
<td>CPU_TIME: 1234.123456</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STMT. BOUND</td>
<td>123456</td>
<td>ELAPSED_TIME: 1234.123456</td>
<td>CPU_TIME: 1234.123456</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STMT. &quot;BOUND</td>
<td>123456</td>
<td>ELAPSED_TIME: 1234.123456</td>
<td>CPU_TIME: 1234.123456</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Here is an example of the Bind Activity Workload Block for Plans.

--- BIND ACTIVITY

<table>
<thead>
<tr>
<th>ISOLATION</th>
<th>CS</th>
<th>TYPE</th>
<th>AUTOMATIC</th>
<th>ACQUIRE</th>
<th>ALLOCATION</th>
<th>DEGREE</th>
<th>ANY</th>
<th>VALIDATE</th>
<th>BIND</th>
</tr>
</thead>
<tbody>
<tr>
<td>OWNER</td>
<td>MANFRED</td>
<td>CURRENTDATA</td>
<td>YES</td>
<td>RELEASE</td>
<td>DEALLOCATION</td>
<td>KEEPDYNAMIC</td>
<td>YES</td>
<td>EXPLAIN</td>
<td>YES</td>
</tr>
<tr>
<td>DYNAMICRULES</td>
<td>BIND</td>
<td>DISCONNECT</td>
<td>CONDITIONAL</td>
<td>PREPARE</td>
<td>NODEFER</td>
<td>QUALIFIER</td>
<td>HUGO</td>
<td>REOPTIMIZE</td>
<td>YES</td>
</tr>
<tr>
<td>ACTION</td>
<td>REPLACE</td>
<td>CACHESIZE</td>
<td>4096</td>
<td>SQLRULES</td>
<td>DB2</td>
<td>PROTOCOL</td>
<td>NOT_SPEC</td>
<td>OPTHINT</td>
<td>YES</td>
</tr>
<tr>
<td>IMMEDIATE</td>
<td>PH1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Field description

Here is a description of the field labels shown in the bind activity workload block for both packages and plans.

**ISOLATION**
Indicates the isolation level of the plan or package.

**TYPE**
The type of bind.

**ACQUIRE**
Indicates when to acquire the locks:
- **ALLOCATION**
  Acquire the locks when the plan or package is allocated.
- **USE**
  Acquire the locks when the application first accesses them.
Utility Activity - Bind Activity

DEGREE
Indicates whether DB2 is to attempt to run a query using parallel processing.
1 Parallelism is prohibited
ANY Parallelism is allowed

VALIDATE
The time of validation:
RUN Validate at run time.
BIND Validate at bind time.

OWNER
The plan or package owner.

CURRENTDATA
Controls the data currency for ambiguous cursors:
NO Data currency is not required for ambiguous cursors. Blocking for ambiguous cursors is allowed.
YES Data currency is required for ambiguous cursors. Blocking for ambiguous cursors is inhibited.
ALL Data currency is required for all cursors. Applicable to packages only.

RELEASE
Indicates when to release the locks:
COMMIT Release locks at commit time.
DEALLOCATION Release locks at deallocation time.

For packages only:
DEFAULT Release locks at run time, which is the default.

KEEPDYNAMIC
Indicates whether the prepared dynamic SQL statements are preserved past a commit:
NO The prepared dynamic SQL statements are destroyed at each commit.
YES The prepared dynamic SQL statements are preserved past a commit. Any subsequent OPEN, EXECUTE, or DESCRIBE assumes that the previous SQL statement is to be executed.

EXPLAIN
Indicates whether EXPLAIN was specified for the bind request.

DYNAMICRULES
The value of the DYNAMICRULES option on the BIND/REBIND command:
RUN run time rules apply to a dynamic SQL statement for authorization checking and object qualification at run time.
**Utility Activity - Bind Activity**

**BIND**  
Bind-time rules apply to a dynamic SQL statement for authorization checking and object qualification at run time.

**N/P**  
DYNAMICRULES was not specified.

**DISCONNECT**  
Indicates which remote connections are terminated during commit operations:

- **EXPLICIT**  
  Only connections in the release state are terminated.

- **AUTOMATIC**  
  All remote connections are terminated.

- **CONDITIONAL**  
  All remote connections are terminated provided that an open WITH HOLD cursor is not associated with the connection.

**PREPARE**  
Indicates whether the preparation of dynamic SQL statements was deferred:

- **DEFER**  
  The preparation of the dynamic SQL statements that refer to remote objects was deferred until run time.

- **NODEFER**  
  The dynamic SQL statements were prepared at bind time.

**QUALIFIER**  
The qualifier used for unqualified object names.

**REOPTIMIZE**  
Indicates whether reoptimization was requested:

- **YES**  
  REOPT(VARS) was specified to reoptimize the access path of the SQL statement at run time.

- **NO**  
  NOREOPT(VARS) was specified to optimize the access path of the SQL statement only at bind time.

**ACTION**  
Specifies whether the plan or package replaces an existing plan or package with the same name or is new:

- **REPLACE**  
  The existing plan or package is replaced.

- **ADD**  
  A new plan or package is added.

This field only applies to BIND activities. For all other activities, N/P is printed.

**IMMEDWRITE**  
Indicates how DB2 updates group buffer pool dependent pages. This is only valid in a data sharing environment.

Group buffer pool dependent pages can be written to DASD or SYSTEM pagesets.

Values shown are:

- **NO**  
  DB2 uses normal write activity for updates, this is the default. Pages are written out at, or before phase 2 commit, or at the end of an abort for transactions that have rolled back.
## Utility Activity - Bind Activity

**PH1** Pages are written out at, or before phase 1 commit.

- If a transaction subsequently rolls back, the pages are updated in the group buffer pool at the end of the rollback and are written out at the end of the abort.

**YES** Pages are written out to the coupling facility as soon as the buffer update commits. Pages are written out regardless of whether the update occurs during forward progress or rollback of the transaction.

- This option can affect performance due to coupling facility overhead.

**SQLERROR**

- Indicates whether a package is created if SQL errors are encountered:
  - CONTINUE A package is created even when SQL errors are encountered.
  - NOPACKAGE No package is created if SQL errors are encountered.

**CACHE SIZE**

- The size (in bytes) of the authorization cache specified for the CACHESIZE keyword. A value of 0 indicates that DB2 determines the size of the authorization cache.

**SQLRULES**

- Indicates whether a type-2 CONNECT statement was executed according to the rules of DB2 or the ISO/ANS SQL2 standard:
  - DB2 An error does not occur if CONNECT identifies an existing SQL statement.
  - STD An error occurs if CONNECT identifies an existing SQL statement.

**PROTOCOL**

- Valid values are:
  - DRDA Protocol is DRDA.
  - PRIVATE Protocol is a private protocol
  - NOT_SPEC Protocol was not specified. This is only valid for packages.

**OPTHINT**

- Indicates whether optimizations hints are to be used. This can be:
  - YES
  - NO

**SENT**

- The number of SQL statements sent to be bound at the server, and the elapsed and CPU times spent for that event at the requester site.

- This field is only shown for remote events.

**STMT. BOUND**

- The number of SQL PARSER events and one or more minibind events that occur between matched BIND or REBIND begin/end record pairs, and the elapsed and CPU times spent for those events.
When a statement is bound, DB2 chooses an access path for the DB2 statement. The only bound DB2 statements are SELECT, UPDATE, INSERT, and DELETE. The other DB2 statements do not require an access path to be generated.

This field shows N/P if the CPU header is not present in the trace data. It is not shown if the IFCIDs 022 and 063 are not available.

**STMT. BOUND**

The number of SQL PARSER events without corresponding minibind events that occur between matched record pairs (BIND or REBIND begin/end), and the elapsed and CPU times spent for those events.

A statement is not bound if DB2 does not calculate an access path.

DECLARE CURSOR and CLOSE CURSOR are examples of statements that are not bound.

This field shows N/P if the CPU header is not present in the trace data. It is not shown if the IFCIDs 022 and 063 are not available.
Data Set Information

This block shows the data set information available for the activity.

Utility Data Set Information Workload Block

Here is an example of the Utility Data Set Information Workload Block.

--- DATA SET INFO -------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**DD NAME** : CCCCCCC8  **DS NAME** : CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC8  **TEMPLATE NAME** : CCCCCCC8
DURATION : 02:03:05.10  DEVICE TYPE : C  OPEN TIMESTAMP: 01/30/10 18:15:44.38

Field description

Here is a description of the field labels shown in the Utility Data Set Information Workload Block:

**DD NAME**
Data definition.

**DS NAME**
Data set name.

**TEMPLATE NAME**
Template name.

**NO.READS**
Number of READ operations.

**NO.WRITES**
Number of WRITE operations.

**NO.CHECKS**
Number of checks.

**NO.EOVS**
Number of End of Volumes.

**I/O WAIT TIME**
I/O wait time.

**DURATION**
The number of seconds the data set was open.

**DEVICE TYPE**
Device type:

D  DASD.
T  Tape.

**OPEN TIMESTAMP**
Time the data set was opened.
Exit Activity

This block shows the exits performed by the event.

Exits Workload Block Example

Here is an example of the Exits Workload Block.

--- EXITS ------------------------------------------------------------------
MEMBER VALIDATION TOTAL AET/EXIT EDIT TOTAL AET/EXIT
SE11 1 N/C 0 0.000060

Field description

Here is a description of the field labels shown in the Exits Workload Block.

MEMBER
The name of the DB2 member within the DB2 data sharing group.

VALIDATION TOTAL
The number of results of a validation exit call written for every validation row.

VALIDATION AET/EXIT
The summarized elapsed validation time divided by the value in VALIDATION TOTAL.

EDIT TOTAL
The summary of results of an edit exit call to encode a record written for every row edited and the results of an edit exit call to decode a record written for every row decoded.

EDIT AET/EXIT
The summarized elapsed edit time divided by the value in EDIT TOTAL.
I/O Activity

This block shows the I/O activity for each object performed by the event.

I/O Activity Workload Block Example

This is an example of the I/O Activity Workload Block.

<table>
<thead>
<tr>
<th>DATABASE</th>
<th>PAGESET</th>
<th>TOTAL</th>
<th>AET</th>
<th>TOTAL</th>
<th>TYPE</th>
<th>AET/WITH</th>
<th>%WITH</th>
<th>PAGE/WITH</th>
<th>%WITHOUT</th>
<th>TOTAL</th>
<th>TYPE</th>
<th>CAST</th>
<th>AET</th>
<th>PAGE/WRIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBPARALL</td>
<td>TSPARALL</td>
<td>3 0.1296</td>
<td>3 SYNCH</td>
<td>0.129597</td>
<td>100.00</td>
<td>1.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WRKSE12</td>
<td>BP0</td>
<td>102 0.0164</td>
<td>102 SYNCH</td>
<td>0.016358</td>
<td>100.00</td>
<td>1.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Field description

Here is a description of the field labels shown in the I/O Activity Workload Block.

DATABASE
The database name. If the name is not available, the decimal DBID/OBID is printed.

PAGESET
The page set name. If the name is not available, the decimal DBID/OBID is printed.

MEMBER
The name of the DB2 member within the DB2 data sharing group.

BP
The buffer pool name.

I/O REQUEST TOTAL
The total number of I/O requests.

I/O REQUEST AET
The average elapsed time for each I/O request.

READ REQUEST TOTAL
The number of read I/O requests of a specific type.

READ REQUEST TYPE
The type of read request:

SYNCH
Synchronous read request

SEQPF
Sequential prefetch request

DYNPF
Dynamic prefetch request

LSTPF
List prefetch request

READ REQUEST AET/WITH
The average elapsed time for a read with I/O of a specific type.

READ REQUEST %WITH
The percentage of total read requests with I/O for a particular type.

READ REQUEST PAGE/WITH
The pages read for each read request with I/O of a particular type.
Utility Activity - I/O

| **READ REQUEST %WITHOUT** | The percentage of total read requests without I/O for a particular type. This can occur because all the pages requested by a prefetch read were already in the buffer pool. |
| **WRITE REQUEST TOTAL** | The number of write I/O requests. |
| **WRITE REQUEST TYPE** | The type of write request. |
| **WRITE REQUEST CAST** | Indicates whether the write operations were initiated due to a coupling facility castout. |
| **WRITE REQUEST AET** | The average elapsed time for each write. |
| **WRITE REQUEST PAGE/WRITE** | The number of pages written. |
LISTDEF Information

This topic describes the workload block of the list definition (LISTDEF) information.

LISTDEF Information Workload Block

This is an example of the workload block of the list definition (LISTDEF) information.

--- LISTDEF LIST INFO --------------------------------------------------------------- ---
LIST NAME : CCCCCCCCCCCCCCCC18 LIST TYPE : M LIST SIZE : 12345

Field description

The workload block of the list definition (LISTDEF) information contains the following fields:

LIST NAME
  Name of list definition information.

LIST TYPE
  Type of LISTDEF information:
  T Table space list.
  I Index space list.
  M Mixed list.
Utility Activity - Lock Suspension

Lock Suspension Activity

This topic shows detailed information about “Workload Detail - Lock Suspension Activity”.

This block shows the lock suspension activity for each object performed by the event.

Lock Suspension Activity Workload Block Example

The field labels shown in the following sample layout of “Lock Suspension Activity Workload Block” are described in the following section.

--- LOCK SUSPENSION ACTIVITY ------------------------------------------------------------------------------------------
---------- SUSPEND REASON ---------- NORML RESUME TIMEO RESUME DEADL RESUME OTHER COUNT AET COUNT AET COUNT AET
RESOURCE NAME TYPE REQUEST LOCAL LATCH IRLMQ GROUP NOTIF OTHER COUNT AET COUNT AET COUNT AET MEMBER
---
DBPARALL TSPARALL DATAPAGE NOTIFY 0 0 0 24 24 0 24 0.74382 0 N/C 0 N/C
SE11
DBPARALL TSPARALL DATAPAGE LOCK 0 3 0 0 0 3 0.04096 0 N/C 0 N/C
SE11
DBPARALL TSPARALL DATAPAGE LOCK 0 5 0 0 0 5 0.06957 0 N/C 0 N/C
SE12
DBPARALL TSPARALL DATAPAGE UNLOCK 0 1 0 2 2 0 3 0.59058 0 N/C 0 N/C
SE21

The following list describes the fields in the lock suspension activity workload block:

**RESOURCE NAME**
The name of the resource on which the suspended request is made. The content of the field depends on the resource type:
- The plan name for SKCT
- The collection and package IDs for SKPT
- The collection ID for COLLECT
- The database name for DATABASE, CDB PLK, DBD PLCK
- The buffer pool ID for ALTERBUF, GBP S/S, P/P PLCK, PAGEPLCK, GBP CAST, P/P CAST
- The anchor point ID for HASH-ANC
- The row ID for ROW
- N/A for MASS, UTILITY, BINDLOCK, ALTERBUF, CATM MIG, CATM CAT, CATM DIR
- The database and page set names for all others

The database and page set names are translations obtained from the IFCIDs 105 and 107. If these records are unavailable, the decimal DBIDs and OBIDs are printed.

**MEMBER**
The name of the DB2 member within the DB2 data sharing group.

**TYPE**
The type of the locked resource. Possible values are shown in Table 33-1 on page 33-3.

**REQUEST**
The type of request that has been suspended:
LOCK  IRLM lock request
UNLOCK  IRLM unlock request
Utility Activity - Lock Suspension

CHANGE
IRLM change request

QUERY
IRLM query request

NOTIFY
IRLM notify request

DRAIN
Drain request

LATCH
Latch request

SUSPEND REASON LOCAL
The number of suspensions due to local resource contentions.

SUSPEND REASON LATCH
The number of suspensions due to IRLM latch contentions.

SUSPEND REASON IRLMQ
The number of suspensions due to IRLM queued requests.

SUSPEND REASON GROUP
The number of suspensions due to global contention.

SUSPEND REASON NOTIFY
The number of suspensions due to intersystem message sending.

SUSPEND REASON OTHER
The number of suspensions due to reasons other than those listed previously.

Note: For drain suspensions, the suspension reason is always “waiting for the claim count to reach zero” and is categorized as OTHER.

NORML RESUME COUNT
The number of suspensions that ended in the task, resuming normal processing after the lock request has completed.

NORML RESUME AET
The normal resume average elapsed time. This is the normal resume elapsed time divided by the NORML RESUME COUNT.

TIMEO RESUME COUNT
The number of suspensions that ended in a timeout.

TIMEO RESUME AET
The average elapsed timeout time. This is the elapsed timeout time divided by the TIMEO RESUME COUNT.

DEADL RESUME COUNT
The number of suspensions that ended in a deadlock.

Note: Drain suspensions do not end in a deadlock.

DEADL RESUME AET
The average elapsed deadlock time. This is the elapsed deadlock time divided by the DEADL RESUME COUNT.
Utility Activity - Page and Row Locking

Page and Row Locking Activity

This block shows the page locking, row locking, and lock avoidance activity for each object, performed by the event.

The page and row locking activity block is only printed if a commit occurred or a thread terminated.

In summary by occurrence, page and row locking activity information generated for explicit commits is shown on the relevant commit events.

In summaries by cursor or program, any explicit commits occurring during the life of that cursor or program are counted. Page and row locking activity caused by those commits is shown on the relevant cursor or program.

In summaries by statement number or statement type, commits are not counted. Because page and row locking activity is not relevant for these summary levels, it is not printed.

Any page or row locking activity occurring when a thread terminated is shown in the summary by thread. This activity is added to any page or row locking which took place in the body of the thread. Therefore, page and row locking figures in summary by thread can be greater than the sum of page locking figures shown in the body of the thread. The difference is the page and row locking activity occurring at thread termination.

An example of the page and row locking workload block is shown in the following example.

Page and Row Locking Workload Block Example

<table>
<thead>
<tr>
<th>MEMBER</th>
<th>DATABASE</th>
<th>PAGESET</th>
<th>COUNT SIZE</th>
<th>OR ROW LOCKS</th>
<th>ESCAL</th>
<th>LOCK</th>
<th>HIGHEST</th>
<th>TS</th>
<th>LOCK AVOID</th>
<th>TYPE</th>
<th>SUCCESSFUL</th>
</tr>
</thead>
<tbody>
<tr>
<td>SE11</td>
<td>DBPARALL</td>
<td>TSPARALL</td>
<td>1 PAGE</td>
<td>1 0</td>
<td>0</td>
<td>SPL</td>
<td>YES</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SUMMARY: MAX PAGE OR ROW LOCKS HELD</td>
<td>1 LOCK ESCALATIONS : SHARED 0 EXCLUSIVE 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SE12</td>
<td>DBPARALL</td>
<td>TSPARALL</td>
<td>2 PAGE</td>
<td>0 0</td>
<td>0</td>
<td>SPL</td>
<td>YES</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SUMMARY: MAX PAGE OR ROW LOCKS HELD</td>
<td>5 LOCK ESCALATIONS : SHARED 0 EXCLUSIVE 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SE21</td>
<td>DBPARALL</td>
<td>TSPARALL</td>
<td>1 PAGE</td>
<td>2 0</td>
<td>0</td>
<td>SPL</td>
<td>YES</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SUMMARY: MAX PAGE OR ROW LOCKS HELD</td>
<td>2 LOCK ESCALATIONS : SHARED 0 EXCLUSIVE 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TOTAL</td>
<td></td>
<td></td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note:

1. The DBID and OBID are obtained from IFCID 020.
2. The values in MAX PAGE OR ROW LOCKS HELD, LOCK ESCALATIONS SHARED, and LOCK ESCALATIONS EXCLUSIVE are accumulated within a subsystem. They are reset only at thread deallocation or when a new user signon occurs.
3. The values in MAXIMUM PAGE OR ROW LOCKS, HIGHEST LOCK, and # LOCK ESCAL are reset at commit time for dynamic BINDs and for static BINDs for which release (commit) is specified. Otherwise, these values accumulate until thread deallocation or until a new user signon occurs.
4. IFCID 218 is an additional lock summary record, written for lock avoidance. It indicates whether a successful lock avoidance test occurred during a given unit of work. The record is externalized for the agent at each commit or rollback.
5. For each event, the relevant IFCID 020 and 218 records are processed. If there is a DBID/OBID combination present for IFCID 218 but not for IFCID 020, the
Utility Activity - Page and Row Locking

IFCID 020 fields show N/P. If there is a DBID/OBID combination present for IFCID 020 but not for IFCID 218, the IFCID 218 field (LOCK AVOID SUCCESSFUL) shows N/P.

Field description

The fields in the page and row locking workload block are:

MEMBER
The name of the DB2 member within the DB2 data sharing group.

DATABASE
The database name, if available.
If the name is not available, the decimal DBID is printed instead.

PAGESET
The page set name, if available.
If the name is not available, the decimal OBID is printed instead.

COUNT
The number of page locking or row locking occurrences for each page set.
  • Specific database and page set:
    – At commit time: always 1
    – At thread termination: the number of times this database and page set occurred on a commit record
  • TOTAL
    – At commit time: the total number of page sets listed
    – At thread termination: the sum of the values for all page sets

LOCK SIZE
The lock size used:
  PAGE  Page lock
  ROW   Row lock
  TABLE  Table space or table lock
  LOB   LOB lock
  UNKN  Unknown lock
  *     Multiple lock sizes

MAXIMUM PAGE OR ROW LOCKS
The maximum number of either page locks or row locks held at one time against this object.

# LOCK ESCAL
The number of lock escalations:
  • 0 if no escalations occur
  • For simple table spaces and partitioned table spaces not using selective partition locking (SPL): 1 if any escalation occurred for this table space in this logical unit of work
  • For segmented table spaces: the number of tables within the table space that have experienced lock escalations
  • For partitioned table spaces using SPL: the number of partitions for which locks escalated within the table space

The TOTAL contains the sum of all values in this column.
Utility Activity - Page and Row Locking

**HIGHEST LOCK**
The highest table space lock state.

If the table space is simple or partitioned not using SPL, it is the highest lock state for this database or page set. At trace end, it is the largest value from any commit for this object. The following values are possible:

- **IS** Intent share
- **IX** Intent exclusive
- **S** Share
- **U** Update share
- **SIX** Share with intent exclusive
- **X** Exclusive

If the table space is segmented or partitioned using SPL, this field is blank.

**TS TYPE**
The table space type:

- **SIMPL** Simple table space
- **SEG** Segmented table space
- **PARTI** Partitioned table space
- **SPL** Partitioned table space using selective partition locking (SPL)
- **LOB** LOB table space

**LOCK AVOID SUCCESSFUL**
Indicates whether there was a successful lock avoidance test during the unit of work.

If the state of this field changed during the summarization period, an asterisk (*) is shown.

**MAX PAGE OR ROW LOCKS HELD**
The maximum number of page locks and row locks held at one time across all objects.

**LOCK ESCALATIONS: SHARED**
The total of shared lock escalations.

**LOCK ESCALATIONS: EXCLUSIVE**
The total of exclusive lock escalations.
Utility Phases

This block shows the utility phases for each object performed by the event. Its layout depends on whether the utility produces parallel tasks.

**Note:** You can process up to 40 utility phases for each object.

### Phases Workload Block Example without Parallel Tasks

Here is an example of the phases workload block without parallel tasks. An example of the phases workload block with parallel tasks is shown in "Phases Workload Block Example with Parallel Tasks."

### Phases Workload Block Example with Parallel Tasks

Here is an example of the phases workload block with parallel tasks.
<table>
<thead>
<tr>
<th>SUBTASK</th>
<th>RUNSTATS</th>
<th>DSNDB04</th>
<th>TSA81401</th>
<th>0</th>
<th>R</th>
<th>12</th>
<th>** SUBTASK TOTAL **</th>
<th>2.125456</th>
<th>0.001180</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.003410</td>
<td>0.000389</td>
<td></td>
</tr>
</tbody>
</table>

** SUBTASK TOTAL ** 2.275380 0.033736

<table>
<thead>
<tr>
<th>SUBTASK</th>
<th>RUNSTATS</th>
<th>DSNDB04</th>
<th>TSA81401</th>
<th>0</th>
<th>R</th>
<th>10</th>
<th>** SUBTASK TOTAL **</th>
<th>2.881214</th>
<th>0.075961</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.000262</td>
<td>0.000182</td>
<td></td>
</tr>
</tbody>
</table>

** SUBTASK TOTAL ** 2.881214 0.075961

** Note:**

1. In LOAD and REORG utility parallelism, the calculation of the elapsed and CPU times for the SORT phase only takes into account the parallel sort, not the originating task.

2. Although not a phase of the LOAD or REORG utility, COPY is reported as a phase when a concurrent COPY was requested for the LOAD or REORG.

3. If the utility runs on several objects or partitions, a TOTAL is shown for each phase.

"Header Fields - Utility Phases” on page 62-19
This topic describes the header fields of the Utility Phases.

"Field description - Utility phases” on page 62-21
This topic describes the fields of the Utility Phases.
Header Fields - Utility Phases

This topic describes the header fields of the Utility Phases.

START TIME
Start of the utility. This is the timestamp of the IFCID 023 (Utility Start) for the UTILINIT.

ELAPSED TIME
Total elapsed time. This is the difference between the timestamp of the IFCID 025 (Utility End) for the UTILTERM and the timestamp of the IFCID 023 (Utility Start) for the UTILINIT.

UTILITY-ID
User–defined utility identifier. This can be up to 16 characters in length.

JOB NAME
User-defined job name.

SHRLEVEL
The SHRLEVEL value of the utility. Possible values are: NONE, REFERENCE, or CHANGE.

SUBTASKS
These three numbers show the following information about parallel subtasks:
1. Requested number of subtasks.
2. Number of actual subtasks.
3. Number of reused subtasks.

N/A is printed for both values when no subtasks or parallelism are used.

CPU TIME
Total CPU time. This is the difference between the CPU time of the IFCID 025 (Utility End) for the UTILTERM and the CPU time of the IFCID 023 (Utility Start) for the UTILINIT.

ACTIVITY TYPE
The name of the activity type and event. The activity type can only be BIND for bind events including remote bind activity, or UTILITY for utility events. All events are indented.

For a detailed description, of Activity Type, see page "Field description" on page 60-1.

DATABASE.PAGESET
The page set ID. This field should match the corresponding field of the preceding IFCID 0024 record.

STEP The step name of the utility job.

TOT. ZIIP
The total utility ZIIP time (if Accounting class 1 trace is activated).

SORT
The type of Sort:
DF DFSORT was invoked at least once. Possible values: are: Y or N.
DB2 DB2 SORT was invoked at least once. Possible values: are: Y or N.
DATA The number of parallel data sorts.
Utility Activity - Workload Header

INDEX
- The number of parallel index sorts.

OTHER
- The number of other sorts.

SORT CPU
- The Sort CPU time.

SORT ZIIP
- The Sort ZIIP time (if provided by the Sort program).
Field description - Utility phases

This topic describes the fields of the Utility Phases.

**PHASE**
The name of the phase used by the utility.

**UNLOAD**
The unload phase of the maintask or the summary of unload subtasks.

**SORT**
The sort phase of the maintask or the summary of sort subtasks.

**BUILD**
The build phase of the maintask or the summary of build subtasks.

**DATABASE**
The database name of the object.

**PAGESET**
The table space name or index name of the object.

When the sort or build phase, or both, are running in parallel as part of a subtask, "*******" is printed if the number of objects is greater than one.

**PARTNO**
The number of the partition or data set if the utility is operating on a single partition or data set. Otherwise, the value in this field is 0.

**TYPE**
The item type for the individual phases.

**COUNT**
The number of item types processed by the phase for one object.

**ELAPSED TIME**
The elapsed time of the phase. This is the time between the IFCID 024 (utility change) of the phase and the IFCID 024 of the next phase. For the last phase, this is the time between the IFCID 024 (utility change) of the phase and the IFCID 024 of the UTILTERM.

**CPU TIME**
The CPU time of the phase. This is the time between the IFCID 024 (utility change) of the phase and the IFCID 024 of the next phase. For the last phase, this is the time between the IFCID 024 (utility change) of the phase and the IFCID 024 of the UTILTERM.

**UTILINIT/ UTILTERM**
This is the starting and ending time of the utility. This is the sum of the time between the IFCID 023 (Utility start) of the UTILINIT and the IFCID 024 (Utility change) of the first phase and the time between the IFCID 024 and the IFCID 025 (Utility end) of the UTILTERM. This is shown as elapsed time and CPU time.

**MAIN TASK TOTAL**
The total time spent processing main tasks. This is shown as elapsed time and CPU time.

**SUBTASK**
For each subtask, the following information is shown:

**SUBTASK**
The time between the IFCID 023 (utility start) for the subtask and the IFCID 024 (utility change) for the first phase within the subtask. This is shown as elapsed time and CPU time.
Utility phases

**Phase**  The name of the phase and time information. For a single phase, this is the time between the IFCID 024 (utility change) for the phase and the IFCID (utility end) of the subtask.

When a subtask contains multiple phases, the duration of the first and intermediate phases is measured from the IFCID 024 of the phase to the IFCID 024 of the next phase. For the last phase in the subtask, phase duration is taken from the IFCID 024 (utility change) for the phase to the IFCID 025 (utility end) of the subtask.

This is shown as elapsed time and CPU time.

**SUBTASK TOTAL**

Total time spent processing the subtask. This is the time between the IFCIDs 23 (utility start) and 25 (utility end) for the subtask.

This is shown as elapsed time and CPU time.
Part 12. Additional Information

These topics provide additional information about reports.

Chapter 63, “The DPMOUT Record,” on page 63-1
The externalized DPMOUT data is a sequential data set with variable-length records. The following table outlines the format of the DPMOUT record.

Chapter 64, “OMEGAMON XE for DB2 PE VSAM Data Sets,” on page 64-1
This section explains the VSAM Data Sets of OMEGAMON XE for DB2 PE.

Chapter 65, “Correlation Translation Record,” on page 65-1
This record layout is not intended to be used as programming interface.

Chapter 66, “Location Information Record,” on page 66-1
This topic describes the record format of the location information.

Chapter 67, “MAINPACK Definitions Record,” on page 67-1
This topic describes the record format of the MAINPACK Definitions.
Chapter 63. The DPMOUT Record

The externalized DPMOUT data is a sequential data set with variable-length records. The following table outlines the format of the DPMOUT record.

Note: Do not use this record as a programming interface.

The DPMOUT record consists of the following sections:
- Header
- Product data section showing:
  - Instrumentation data
  - CPU data
  - DDF data
  - Data sharing information
- Repeating section information
- DBID and OBID translation information

The following tables show the layout of the DPMOUT record:
- DPMOUT header

Table 63-1. Layout of the DPMOUT Record (DPMOUT header)

<table>
<thead>
<tr>
<th>Offset</th>
<th>Length</th>
<th>Data Type</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2</td>
<td>FIXED</td>
<td>Record length</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>FIXED</td>
<td>Reserved (zeros)</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>CHAR</td>
<td>'DPM'</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>FIXED</td>
<td>DB2PM version release flag</td>
</tr>
<tr>
<td>8</td>
<td>4</td>
<td>FIXED</td>
<td>Full record length</td>
</tr>
<tr>
<td>12</td>
<td>60</td>
<td>CHAR</td>
<td>SORT Key</td>
</tr>
<tr>
<td>12</td>
<td>16</td>
<td>CHAR</td>
<td>Location (EBCDIC)</td>
</tr>
<tr>
<td>28</td>
<td>8</td>
<td>CHAR</td>
<td>Group name</td>
</tr>
<tr>
<td>36</td>
<td>4</td>
<td>CHAR</td>
<td>Subsystem identifier</td>
</tr>
<tr>
<td>40</td>
<td>8</td>
<td>CHAR</td>
<td>Member</td>
</tr>
<tr>
<td>48</td>
<td>8</td>
<td>CHAR</td>
<td>SORT timestamp</td>
</tr>
<tr>
<td>56</td>
<td>1</td>
<td>CHAR</td>
<td>Destination code</td>
</tr>
<tr>
<td>57</td>
<td>4</td>
<td>CHAR</td>
<td>Destination sequence number</td>
</tr>
<tr>
<td>61</td>
<td>2</td>
<td>CHAR</td>
<td>Split record sequence no.</td>
</tr>
<tr>
<td>63</td>
<td>9</td>
<td>CHAR</td>
<td>Reserved</td>
</tr>
<tr>
<td>72</td>
<td>1</td>
<td>BIT 0</td>
<td>Record processing flags</td>
</tr>
<tr>
<td>73</td>
<td>3</td>
<td>CHAR</td>
<td>Reserved</td>
</tr>
<tr>
<td>76</td>
<td>8</td>
<td>CHAR</td>
<td>TIMEZONE adjusted timestamp</td>
</tr>
<tr>
<td>84</td>
<td>12</td>
<td>CHAR</td>
<td>Correlation name (translated)</td>
</tr>
<tr>
<td>96</td>
<td>8</td>
<td>CHAR</td>
<td>Correlation number (translated)</td>
</tr>
<tr>
<td>104</td>
<td>8</td>
<td>CHAR</td>
<td>Connecting system type</td>
</tr>
<tr>
<td>112</td>
<td>1</td>
<td>BIT 0</td>
<td>Record type flag</td>
</tr>
<tr>
<td>113</td>
<td>1</td>
<td>CHAR</td>
<td>Correlation data present</td>
</tr>
</tbody>
</table>
Table 63-1. Layout of the DPMOUT Record (DPMOUT header) (continued)

<table>
<thead>
<tr>
<th>Offset</th>
<th>Length</th>
<th>Data Type</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>114</td>
<td>1</td>
<td>CHAR</td>
<td>CPU data present</td>
</tr>
<tr>
<td>115</td>
<td>1</td>
<td>CHAR</td>
<td>DDF data present</td>
</tr>
<tr>
<td>116</td>
<td>4</td>
<td>PTR</td>
<td>Offset to DBID/OBID section</td>
</tr>
<tr>
<td>120</td>
<td>2</td>
<td>FIXED</td>
<td>Length to DBID/OBID section</td>
</tr>
<tr>
<td>122</td>
<td>2</td>
<td>FIXED</td>
<td>Number to DBID/OBID section</td>
</tr>
<tr>
<td>124</td>
<td>2</td>
<td>FIXED</td>
<td>Offset to DBID/OBID strings</td>
</tr>
<tr>
<td>126</td>
<td>2</td>
<td>FIXED</td>
<td>length of DBID/OBID strings</td>
</tr>
<tr>
<td>128</td>
<td>2</td>
<td>FIXED</td>
<td>Total no. of split records</td>
</tr>
<tr>
<td>130</td>
<td>2</td>
<td>FIXED</td>
<td>Offset to long identifier</td>
</tr>
<tr>
<td>132</td>
<td>2</td>
<td>FIXED</td>
<td>Length of DPM0 header and</td>
</tr>
<tr>
<td>134</td>
<td>2</td>
<td>FIXED</td>
<td>Reserved</td>
</tr>
</tbody>
</table>

* Product Data, Instrumentation Data

Table 63-2. Layout of the DPMOUT Record (Product Data, Instrumentation Data)

<table>
<thead>
<tr>
<th>Offset</th>
<th>Length</th>
<th>Data Type</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>136</td>
<td>3</td>
<td>CHAR</td>
<td>Reserved</td>
</tr>
<tr>
<td>139</td>
<td>1</td>
<td>FIXED</td>
<td>Resource manager id</td>
</tr>
<tr>
<td>140</td>
<td>2</td>
<td>FIXED</td>
<td>IFCID</td>
</tr>
<tr>
<td>142</td>
<td>1</td>
<td>FIXED</td>
<td>Self defining area count</td>
</tr>
<tr>
<td>143</td>
<td>1</td>
<td>FIXED</td>
<td>DB2 version/release</td>
</tr>
<tr>
<td>144</td>
<td>4</td>
<td>PTR</td>
<td>ACE address</td>
</tr>
<tr>
<td>148</td>
<td>4</td>
<td>CHAR</td>
<td>Subsystem name</td>
</tr>
<tr>
<td>152</td>
<td>8</td>
<td>CHAR</td>
<td>Store clock value of header</td>
</tr>
<tr>
<td>160</td>
<td>4</td>
<td>FIXED</td>
<td>IFCID sequence number</td>
</tr>
<tr>
<td>164</td>
<td>4</td>
<td>FIXED</td>
<td>Destination sequence number</td>
</tr>
<tr>
<td>168</td>
<td>4</td>
<td>FIXED</td>
<td>Active trace number mask</td>
</tr>
<tr>
<td>172</td>
<td>16</td>
<td>CHAR</td>
<td>Local location name</td>
</tr>
<tr>
<td>188</td>
<td>24</td>
<td>CHAR</td>
<td>Logical unit of work ID (LUWID)</td>
</tr>
<tr>
<td>188</td>
<td>8</td>
<td>CHAR</td>
<td>Net id</td>
</tr>
<tr>
<td>196</td>
<td>8</td>
<td>CHAR</td>
<td>LU name</td>
</tr>
<tr>
<td>204</td>
<td>6</td>
<td>CHAR</td>
<td>Instance number</td>
</tr>
<tr>
<td>210</td>
<td>2</td>
<td>CHAR</td>
<td>Commit count</td>
</tr>
<tr>
<td>212</td>
<td>1</td>
<td>BIT 0</td>
<td>QWHS_FLAGS</td>
</tr>
<tr>
<td>212</td>
<td>1</td>
<td>BIT 1</td>
<td>Reserved</td>
</tr>
<tr>
<td>213</td>
<td>1</td>
<td>BIT 0</td>
<td>Reserved</td>
</tr>
<tr>
<td>214</td>
<td>2</td>
<td>FIXED</td>
<td>Offset to long location</td>
</tr>
<tr>
<td>216</td>
<td>2</td>
<td>FIXED</td>
<td>Record sub-version</td>
</tr>
<tr>
<td>218</td>
<td>2</td>
<td>CHAR</td>
<td>Reserved</td>
</tr>
</tbody>
</table>

* Product Data, Correlation Data
Table 63-3. Layout of the DPMOUT Record (Product Data, Correlation Data)

<table>
<thead>
<tr>
<th>Offset</th>
<th>Length</th>
<th>Data Type</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>220</td>
<td>8</td>
<td>CHAR</td>
<td>Authorization id</td>
</tr>
<tr>
<td>228</td>
<td>12</td>
<td>CHAR</td>
<td>Correlation id</td>
</tr>
<tr>
<td>240</td>
<td>8</td>
<td>CHAR</td>
<td>Connection id</td>
</tr>
<tr>
<td>248</td>
<td>8</td>
<td>CHAR</td>
<td>Plan name</td>
</tr>
<tr>
<td>256</td>
<td>8</td>
<td>CHAR</td>
<td>Original operator id</td>
</tr>
<tr>
<td>264</td>
<td>4</td>
<td>FIXED</td>
<td>Connecting system type code</td>
</tr>
<tr>
<td>268</td>
<td>22</td>
<td>CHAR</td>
<td>Accounting token</td>
</tr>
<tr>
<td>290</td>
<td>2</td>
<td>CHAR</td>
<td>Reserved</td>
</tr>
<tr>
<td>292</td>
<td>66</td>
<td>CHAR</td>
<td>End user workstation data</td>
</tr>
<tr>
<td>308</td>
<td>16</td>
<td>CHAR</td>
<td>End user workstation userID</td>
</tr>
<tr>
<td>320</td>
<td>32</td>
<td>CHAR</td>
<td>End user workstation trans.</td>
</tr>
<tr>
<td>336</td>
<td>18</td>
<td>CHAR</td>
<td>End user workstation name</td>
</tr>
<tr>
<td>358</td>
<td>2</td>
<td>FIXED</td>
<td>Offset to long auth id</td>
</tr>
<tr>
<td>360</td>
<td>2</td>
<td>FIXED</td>
<td>Offset to long oper id</td>
</tr>
<tr>
<td>362</td>
<td>2</td>
<td>FIXED</td>
<td>Offset to long euser id</td>
</tr>
</tbody>
</table>

- Product Data, CPU Data

Table 63-4. Layout of the DPMOUT Record (Product Data, CPU Data)

<table>
<thead>
<tr>
<th>Offset</th>
<th>Length</th>
<th>Data Type</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>364</td>
<td>8</td>
<td>CHAR</td>
<td>CPU time</td>
</tr>
<tr>
<td>372</td>
<td>2</td>
<td>FIXED</td>
<td>Count field reserved (s)</td>
</tr>
<tr>
<td>374</td>
<td>2</td>
<td>FIXED</td>
<td>Reserved</td>
</tr>
</tbody>
</table>

- Product Data, DDF Data

Table 63-5. Layout of the DPMOUT Record (Product Data, DDF Data)

<table>
<thead>
<tr>
<th>Offset</th>
<th>Length</th>
<th>Data Type</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>376</td>
<td>16</td>
<td>CHAR</td>
<td>Requester location name</td>
</tr>
<tr>
<td>392</td>
<td>8</td>
<td>CHAR</td>
<td>STCK for DBAT trace records</td>
</tr>
<tr>
<td>400</td>
<td>16</td>
<td>CHAR</td>
<td>Server name</td>
</tr>
<tr>
<td>416</td>
<td>8</td>
<td>CHAR</td>
<td>PRDID parm (DB2 2.3)</td>
</tr>
<tr>
<td>424</td>
<td>2</td>
<td>FIXED</td>
<td>Offset to requester name</td>
</tr>
<tr>
<td>426</td>
<td>2</td>
<td>FIXED</td>
<td>Offset to server name</td>
</tr>
</tbody>
</table>

- Product Data, Data Sharing

Table 63-6. Layout of the DPMOUT Record (Product Data, Data Sharing)

<table>
<thead>
<tr>
<th>Offset</th>
<th>Length</th>
<th>Data Type</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>428</td>
<td>8</td>
<td>CHAR</td>
<td>DB2 Member name</td>
</tr>
<tr>
<td>436</td>
<td>8</td>
<td>CHAR</td>
<td>DB2 data sharing group</td>
</tr>
<tr>
<td>Offset</td>
<td>Length</td>
<td>Data Type</td>
<td>Field Description</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>-----------</td>
<td>------------------------</td>
</tr>
<tr>
<td>444</td>
<td>4</td>
<td>PTR</td>
<td>Data section 1 offset</td>
</tr>
<tr>
<td>448</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 1 length</td>
</tr>
<tr>
<td>450</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 1 count</td>
</tr>
<tr>
<td>452</td>
<td>4</td>
<td>PTR</td>
<td>Data section 2 offset</td>
</tr>
<tr>
<td>456</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 2 length</td>
</tr>
<tr>
<td>458</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 2 count</td>
</tr>
<tr>
<td>460</td>
<td>4</td>
<td>PTR</td>
<td>Data section 3 offset</td>
</tr>
<tr>
<td>464</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 3 length</td>
</tr>
<tr>
<td>466</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 3 count</td>
</tr>
<tr>
<td>468</td>
<td>4</td>
<td>PTR</td>
<td>Data section 4 offset</td>
</tr>
<tr>
<td>472</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 4 length</td>
</tr>
<tr>
<td>474</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 4 count</td>
</tr>
<tr>
<td>476</td>
<td>4</td>
<td>PTR</td>
<td>Data section 5 offset</td>
</tr>
<tr>
<td>480</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 5 length</td>
</tr>
<tr>
<td>482</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 5 count</td>
</tr>
<tr>
<td>484</td>
<td>4</td>
<td>PTR</td>
<td>Data section 6 offset</td>
</tr>
<tr>
<td>488</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 6 length</td>
</tr>
<tr>
<td>490</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 6 count</td>
</tr>
<tr>
<td>492</td>
<td>4</td>
<td>PTR</td>
<td>Data section 7 offset</td>
</tr>
<tr>
<td>496</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 7 length</td>
</tr>
<tr>
<td>498</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 7 count</td>
</tr>
<tr>
<td>500</td>
<td>4</td>
<td>PTR</td>
<td>Data section 8 offset</td>
</tr>
<tr>
<td>504</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 8 length</td>
</tr>
<tr>
<td>506</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 8 count</td>
</tr>
<tr>
<td>508</td>
<td>4</td>
<td>PTR</td>
<td>Data section 9 offset</td>
</tr>
<tr>
<td>512</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 9 length</td>
</tr>
<tr>
<td>514</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 9 count</td>
</tr>
<tr>
<td>516</td>
<td>4</td>
<td>PTR</td>
<td>Data section 10 offset</td>
</tr>
<tr>
<td>520</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 10 length</td>
</tr>
<tr>
<td>522</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 10 count</td>
</tr>
<tr>
<td>524</td>
<td>4</td>
<td>PTR</td>
<td>Data section 11 offset</td>
</tr>
<tr>
<td>528</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 11 length</td>
</tr>
<tr>
<td>530</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 11 count</td>
</tr>
<tr>
<td>532</td>
<td>4</td>
<td>PTR</td>
<td>Data section 12 offset</td>
</tr>
<tr>
<td>536</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 12 length</td>
</tr>
<tr>
<td>538</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 12 count</td>
</tr>
<tr>
<td>540</td>
<td>4</td>
<td>PTR</td>
<td>Data section 13 offset</td>
</tr>
<tr>
<td>544</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 13 length</td>
</tr>
<tr>
<td>546</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 13 count</td>
</tr>
<tr>
<td>548</td>
<td>4</td>
<td>PTR</td>
<td>Data section 14 offset</td>
</tr>
</tbody>
</table>
### Table 63-7. Layout of the DPMOUT Record (Self-Defining Sections) (continued)

<table>
<thead>
<tr>
<th>Offset</th>
<th>Length</th>
<th>Data Type</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>552</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 14 length</td>
</tr>
<tr>
<td>554</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 14 count</td>
</tr>
<tr>
<td>556</td>
<td>4</td>
<td>PTR</td>
<td>Data section 15 offset</td>
</tr>
<tr>
<td>560</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 15 length</td>
</tr>
<tr>
<td>562</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 15 count</td>
</tr>
<tr>
<td>564</td>
<td>4</td>
<td>PTR</td>
<td>Data section 16 offset</td>
</tr>
<tr>
<td>568</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 16 length</td>
</tr>
<tr>
<td>570</td>
<td>2</td>
<td>FIXED</td>
<td>Data section 16 count</td>
</tr>
<tr>
<td>572</td>
<td>2</td>
<td>FIXED</td>
<td>Reserved</td>
</tr>
</tbody>
</table>

### DBID/OBID Translation

**Table 63-8. Layout of the DPMOUT Record (DBID/OBID Translation)**

<table>
<thead>
<tr>
<th>Offset</th>
<th>Length</th>
<th>Data Type</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2</td>
<td>FIXED</td>
<td>Offset to DBID/OBID section</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>FIXED</td>
<td>OBID number</td>
</tr>
<tr>
<td>4</td>
<td>8</td>
<td>CHAR</td>
<td>Database name</td>
</tr>
<tr>
<td>12</td>
<td>8</td>
<td>CHAR</td>
<td>Table or index space name</td>
</tr>
<tr>
<td>20</td>
<td>2</td>
<td>FIXED</td>
<td>Offset to index name</td>
</tr>
<tr>
<td>22</td>
<td>1</td>
<td>CHAR</td>
<td>Pageset type: I,T,U</td>
</tr>
<tr>
<td>23</td>
<td>1</td>
<td>CHAR</td>
<td>Reserved</td>
</tr>
<tr>
<td>24</td>
<td>4</td>
<td>FIXED</td>
<td>Object size in pages</td>
</tr>
</tbody>
</table>
Chapter 64. OMEGAMON XE for DB2 PE VSAM Data Sets

This section explains the VSAM Data Sets of OMEGAMON XE for DB2 PE.

OMEGAMON XE for DB2 PE uses the following VSAM data sets:
- A VSAM-Save data set is written when the job stream contains a SAVE subcommand without the CONVERT option.
- A physical sequential data set is written when the job stream contains a SAVE subcommand with a CONVERT option.
- Job summary data sets are written when new data is processed.

All VSAM data sets used in an OMEGAMON XE for DB2 PE job must exist before OMEGAMON XE for DB2 PE is executed. Preallocate the data sets using the IDCAMS command. You can run IDCAMS as an initial step in the OMEGAMON XE for DB2 PE job. The required attributes for VSAM data sets are shown in Table 64-1. An example of the required IDCAMS commands is shown in "IDCAMS Commands."

Refer to the z/OS DFSMS Access Method Services for Catalogs for more information about IDCAMS.

Note:
1. When the SAVE subcommand is specified, the save data set should be empty. If it is not empty, all existing records are deleted. If save and restore use the same physical data set, the restored data is rewritten during save.
2. You need not prime OMEGAMON XE for DB2 PE VSAM data sets.

Table 64-1. Attributes for OMEGAMON XE for DB2 PE VSAM Data Sets

<table>
<thead>
<tr>
<th>Data Set</th>
<th>Key Length (bytes)</th>
<th>Maximum Record Length (bytes)</th>
<th>Average Record Length (bytes)</th>
<th>Buffer Space (bytes)</th>
<th>Data Control Interval Size (bytes)</th>
<th>Index Control Interval Size (bytes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accounting SAVE (ACSAVDD)</td>
<td>255</td>
<td>5400</td>
<td>2800</td>
<td>40 960</td>
<td>8192</td>
<td>4096</td>
</tr>
<tr>
<td>Statistics SAVE (STSAVDD)</td>
<td>92</td>
<td>8192</td>
<td>2400</td>
<td>40 960</td>
<td>8192</td>
<td>4096</td>
</tr>
<tr>
<td>Job Summary (JSSRSDD)</td>
<td>52</td>
<td>2462</td>
<td>160</td>
<td>40 960</td>
<td>8192</td>
<td>4096</td>
</tr>
</tbody>
</table>

Note: Buffer space and control interval size are suggestions only. You can modify them to suit the requirements of your installation.

IDCAMS Commands

In this example, the job deletes the cluster if it already exists, then defines a new cluster with the specified attributes.

//ALCVSAM EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*  
//SYSIN DD *
DELETE (cluster.name)
DEFINE -
   CLUSTER { -

© Copyright IBM Corp. 1985, 2013
Chapter 65. Correlation Translation Record

This record layout is not intended to be used as programming interface.

Table 65-1. Correlation Translation Record

<table>
<thead>
<tr>
<th>Offset</th>
<th>Length</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>8</td>
<td>Connection ID</td>
</tr>
<tr>
<td>8</td>
<td>2</td>
<td>Correlation name offset</td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>Correlation name length</td>
</tr>
<tr>
<td>12</td>
<td>2</td>
<td>Correlation number offset</td>
</tr>
<tr>
<td>14</td>
<td>2</td>
<td>Correlation number length</td>
</tr>
<tr>
<td>16</td>
<td>64</td>
<td>Reserved</td>
</tr>
</tbody>
</table>
Chapter 66. Location Information Record

This topic describes the record format of the location information.

**Table 66-1. Location Information Record Format**

<table>
<thead>
<tr>
<th>Offset</th>
<th>Length</th>
<th>Data Type</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>16</td>
<td>CHAR</td>
<td>Location</td>
</tr>
<tr>
<td>16</td>
<td>2</td>
<td>CHAR</td>
<td>Reserved</td>
</tr>
<tr>
<td>18</td>
<td>1</td>
<td>CHAR</td>
<td>Local time relativity (E or W)</td>
</tr>
<tr>
<td>19</td>
<td>5</td>
<td>CHAR</td>
<td>Difference between local time and GMT (HH:MM)</td>
</tr>
<tr>
<td>24</td>
<td>1</td>
<td>CHAR</td>
<td>CPU time relativity (E or W)</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
<td>CHAR</td>
<td>Difference between CPU time and GMT (HH:MM)</td>
</tr>
<tr>
<td>30</td>
<td>50</td>
<td>CHAR</td>
<td>Reserved</td>
</tr>
</tbody>
</table>
Chapter 67. MAINPACK Definitions Record

This topic describes the record format of the MAINPACK Definitions.

Note: This record layout is not intended to be used as programming interface.

<table>
<thead>
<tr>
<th>Offset</th>
<th>Length</th>
<th>Data Type</th>
<th>Field Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>16</td>
<td>CHAR</td>
<td>Requester location</td>
</tr>
<tr>
<td>16</td>
<td>8</td>
<td>CHAR</td>
<td>Connection ID</td>
</tr>
<tr>
<td>24</td>
<td>8</td>
<td>CHAR</td>
<td>Plan name</td>
</tr>
<tr>
<td>32</td>
<td>1</td>
<td>CHAR</td>
<td>Code</td>
</tr>
</tbody>
</table>
Part 13. Appendixes
Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.
Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Deutschland GmbH
Dept. M358
IBM-Allee 1
71139 Ehningen
Germany

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Adobe is either a registered trademark or a trademark of Adobe Systems Incorporated in the United States, and/or other countries.

Intel, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other product and service names might be trademarks of IBM or other companies.
Bibliography

You can order many IBM publications such as product manuals or IBM Redbooks online at the IBM Publications Center website.

You can also order by telephone by calling one of the following numbers:

- In the United States: 800-879-2755
- In Canada: 800-426-4968

In other countries, contact your software account representative to order Tivoli publications.

**IBM Tivoli OMEGAMON XE for DB2 Performance Expert publications**

The product library for Version 5 Release 2 covers the following information units:

**OMEGAMON XE for DB2 PE and OMEGAMON XE for DB2 PM**

- Configuration and Customization, GH12-6998
- Parameter Reference, SH12-6999
- Monitoring Performance from ISPF, SH12-6996
- Monitoring Performance from the OMEGAMON Classic Interface, SH12-6994
- Monitoring Performance from Performance Expert Client, SH12-6995
- Report Command Reference, SH12-6992
- Report Reference, SH12-6991
- Reporting User’s Guide, SH12-6997
- Messages, GH12-6993
- Program Directory for Performance Monitor, GI19-5009
- Program Directory for Performance Expert, GI19-5007
- Quick Start Guide for the end-to-end SQL monitoring function, GH12-6990

**Buffer Pool Analyzer**

- Buffer Pool Analyzer Configuration Guide, SH12-7030
- Program Directory for IBM DB2 Buffer Pool Analyzer for z/OS, GI19-5010

**InfoSphere Optim Performance Manager for Linux, UNIX, and Windows**

- InfoSphere® Optim™ Performance Manager Installation Guide, GC19-2934

The documentation is provided in PDF and htm format in the:

- [Tivoli OMEGAMON XE for DB2 Performance Expert on z/OS information center](#)
- [Tivoli OMEGAMON XE for DB2 Performance Monitor on z/OS information center](#)

**IBM Tivoli Monitoring publications**

For the most current list of publications, see the [Tivoli Monitoring information center](#)
IBM DB2 publications

For the most current list of publications, see the IBM Information Management Software for z/OS Solutions information center.

Other IBM publications

For IBM publications that are not directly related to OMEGAMON XE for DB2 PE and PM, see Related information in the MVS Programming Hiperbatch Guide.
Index

A

Accelerator report and trace blocks 5-48
Accounting 49-5
access path data explain 14-1
accessibility features xxx
accounting averages 5-4
headers 5-2
long report 5-35
long trace 5-39
short report 5-6
short trace 5-20
Accounting report and trace blocks (continued)
  Package General (Short Trace) 5-28
  Package Global Contention L-Locks - Class 8 5-132
  Package Global Contention P-Locks - Class 8 5-134
  Package Identification 5-136
  Package Identification - Report 5-137
  Package Identification - Trace 5-142
  Package Locking Activity - Class 10 5-147
  Package SQL Activity - Class 10 5-150
  Package Times - Class 7 5-160
  Package Times - Class 8 - Suspensions 5-152
  Query Parallelism 5-163
  Resource Limit Facility 5-167
  RID List 5-169
  ROWID 5-172
  Service Units 5-173
  SQL DCL 5-179
  SQL DDL 5-182
  SQL DML 5-188
  Stored Procedures 5-178
  Termination - Abnormal 5-191
  Termination - In Doubt 5-192
  Termination - Normal 5-193
  Times - Class 1 - Application Time 5-195
  Times - Class 1 - Elapsed Time Distribution 5-201
  Times - Class 2 - DB2 Time 5-202
  Times - Class 2 - Time Distribution 5-209
  Times - Class 3 - Suspensions 5-210
  Times - Class 5 - IFI Time 5-219
  Times - Class 7 - CP CPU Distribution 5-221
  Times - Class 7 - Elapsed Time Distribution 5-222
  Triggers 5-224
  Truncated Values 5-225
  User-Defined Functions 5-226
  trace blocks
  Initial DB2 Common Server Traces 5-112
  Accounting report default layout 5-1
  introduction 0-7
  Accounting Rollup report and trace blocks
  Statistics 49-9
  Aggregated Accounting Statistics report and trace blocks
  Statistics 49-10
  Alter Buffer Pool Command Issued report and trace blocks
  System Parameters 55-1
  Alter Group Buffer Pool Command Issued report and trace blocks
  System Parameters 56-1
  Application Programming Defaults Panel 1 (DSNTIPF) report and trace blocks
  System Parameters 54-3
  Application Programming Defaults Panel 2 (DSNTIP4, DSNTIP41) report and trace blocks
  System Parameters 54-9
  Archive Log Installation Parameters (DSNTIPA) report and trace blocks
  System Parameters 54-13
  audit
    authorization change detail (Type AUTHCHG) 9-12
    authorization change summary (AUTHCHG) 8-5
    authorization control detail (type AUTHCNTL) 9-22
    authorization control summary (AUTHCNTL) 8-7
    authorization failure detail (type AUTHFAIL) 9-26
    authorization failure summary (AUTHFAIL) 8-9
    DDL access detail (DDL) 9-30
    DDL access summary (DDL) 8-14
detail 9-1
    DML access detail (type DML) 9-34
    DML access summary (DML) 8-16
    DML at bind access detail (type BIND) 9-28
    DML at bind access summary (BIND) 8-12
    file data set 10-1
    summary report 8-1
    trace 9-1
    utility access detail (Type UTILITY) 9-35
    utility access summary (UTILITY) 8-18
    audit report 0-3
    authorization change detail (Type AUTHCHG) audit 9-12
    authorization change summary (AUTHCHG) audit 8-5
    authorization control detail (type AUTHCNTL) audit 9-22
    authorization control summary (AUTHCNTL) audit 8-7
IFCID 003 - Data Sharing Locking Data report and trace blocks
Record Trace 40-154
IFCID 003 - DDF Data by Location report and trace blocks
Record Trace 40-156
IFCID 003 - Dynamic SQL Statement report and trace blocks
Record Trace 40-165
IFCID 003 - Group Buffer Pools Activity Data report and trace blocks
Record Trace 40-167
IFCID 003 - IFI Class 5 Times and Data Capture report and trace blocks
Record Trace 40-171
IFCID 003 - Instrumentation Accounting Data report and trace blocks
Record Trace 40-175
IFCID 003 - Instrumentation Accounting Data Overflow report and trace blocks
Record Trace 40-190
IFCID 003 -Locking Data report and trace blocks
Record Trace 40-193
IFCID 003 - Logging report and trace blocks
Record Trace 40-197
IFCID 003 - Miscellaneous report and trace blocks
Record Trace 40-198
IFCID 003 - Nested SQL Activity report and trace blocks
Record Trace 40-199
IFCID 003 - Query Parallelism report and trace blocks
Record Trace 40-201
IFCID 003 - Resource Limit Facility report and trace blocks
Record Trace 40-202
IFCID 003 - RID List Processing report and trace blocks
Record Trace 40-204
IFCID 003 - Rollup Accounting Correlation Block (DB2 10 or later) report and trace blocks
Record Trace 40-205
IFCID 003 - ROWID report and trace blocks
Record Trace 40-206
IFCID 003 - SQL Call Data report and trace blocks
Record Trace 40-207
IFCID 004 - Trace Start report and trace blocks
Record Trace 40-208
IFCID 005 - Trace Stop report and trace blocks
Record Trace 40-209
IFCID 006 - Read I/O Start report and trace blocks
Record Trace 40-210
IFCID 007 - Read I/O Stop report and trace blocks
Record Trace 40-212
IFCID 008 - Write I/O Start report and trace blocks
Record Trace 40-213
IFCID 009 - Write I/O report and trace blocks
Record Trace 40-215
IFCID 011 - Validate Exit report and trace blocks
Record Trace 40-216
IFCID 012 - Edit Exit to Encode report and trace blocks
Record Trace 40-219
IFCID 013 - Hash Scan Input Start report and trace blocks
Record Trace 40-220
IFCID 014 - Hash Scan End report and trace blocks
Record Trace 40-222
IFCID 015 - Index Scan Begin report and trace blocks
Record Trace 40-223
IFCID 016 - Insert Scan Begin report and trace blocks
Record Trace 40-225
IFCID 017 - Sequential Scan Begin report and trace blocks
Record Trace 40-227
IFCID 018 - Scan End report and trace blocks
Record Trace 40-229
IFCID 019 - Edit Exit to Decode report and trace blocks
Record Trace 40-231
IFCID 020 - Lock Summary report and trace blocks
Record Trace 40-232
IFCID 021 - Lock Detail report and trace blocks
Record Trace 40-234
IFCID 022 - Minibind report and trace blocks
Record Trace 40-239
IFCID 023 - Utility Start report and trace blocks
Record Trace 40-248
IFCID 024 - Utility Change report and trace blocks
Record Trace 40-252
IFCID 025 - Utility End report and trace blocks
Record Trace 40-254
IFCID 026 - IBM Service Record report and trace blocks
Record Trace 40-257
IFCID 027 - Sort Workfile Records report and trace blocks
Record Trace 40-258
IFCID 028 - Sort Phase Detail report and trace blocks
Record Trace 40-260
IFCID 029 - EDM Request Start report and trace blocks
Record Trace 40-262
IFCID 030 - EDM Request End report and trace blocks
Record Trace 40-264
IFCID 031 - EDM Full report and trace blocks
Record Trace 40-266
IFCID 032 - Log Wait Start report and trace blocks
Record Trace 40-269
IFCID 033 - IBM Service Record report and trace blocks
Record Trace 40-270
IFCID 034 - Log Read Start report and trace blocks
Record Trace 40-271
IFCID 035 - Log Read End report and trace blocks
Record Trace 40-272
IFCID 036 - Log Non I/O Start report and trace blocks
Record Trace 40-273
IFCID 037 - Log Non I/O End report and trace blocks
Record Trace 40-274
IFCID 038 - Active Write Start report and trace blocks
Record Trace 40-275
IFCID 039 - Active Write End report and trace blocks
Record Trace 40-276
IFCID 040 - Archive Write Start report and trace blocks
Record Trace 40-277
IFCID 041 - Archive Write End report and trace blocks
Record Trace 40-278
IFCID 042 - Checkpoint Start report and trace blocks
Record Trace 40-279
IFCID 043 - Checkpoint End report and trace blocks
Record Trace 40-280
IFCID 044 - Lock Suspend report and trace blocks
Record Trace 40-281
IFCID 045 - Lock Resume report and trace blocks
Record Trace 40-282
IFCID 046 - IBM Service Record report and trace blocks
Record Trace 40-287
IFCID 047 - IBM Service Record report and trace blocks
Record Trace 40-288
IFCID 048 - IBM Service Record report and trace blocks
Record Trace 40-289
IFCID 049 - IBM Service Record report and trace blocks
Record Trace 40-290
<table>
<thead>
<tr>
<th>IFCID 050 - IBM Service Record</th>
<th>IFCID 073 - Create Thread End</th>
<th>IFCID 095 - Sort Start</th>
</tr>
</thead>
<tbody>
<tr>
<td>report and trace blocks</td>
<td>report and trace blocks</td>
<td>report and trace blocks</td>
</tr>
<tr>
<td>Record Trace 40-291</td>
<td>Record Trace 40-325</td>
<td>Record Trace 40-347</td>
</tr>
<tr>
<td>IFCID 051 - IBM Service Record</td>
<td>IFCID 074 - Terminate Thread Start</td>
<td>IFCID 096 - Sort End</td>
</tr>
<tr>
<td>report and trace blocks</td>
<td>report and trace blocks</td>
<td>report and trace blocks</td>
</tr>
<tr>
<td>Record Trace 40-292</td>
<td>Record Trace 40-326</td>
<td>Record Trace 40-348</td>
</tr>
<tr>
<td>IFCID 052 - IBM Service Record</td>
<td>IFCID 075 - Terminate Thread End</td>
<td>IFCID 097 - AMS Command End</td>
</tr>
<tr>
<td>report and trace blocks</td>
<td>report and trace blocks</td>
<td>report and trace blocks</td>
</tr>
<tr>
<td>Record Trace 40-293</td>
<td>Record Trace 40-327</td>
<td>Record Trace 40-351</td>
</tr>
<tr>
<td>IFCID 053 - SQL Describe/Commit/Rollback/Remote Statement report and trace blocks</td>
<td>IFCID 076 - End of Memory Start</td>
<td>IFCID 098 - IBM Service Record report and trace blocks</td>
</tr>
<tr>
<td>Record Trace 40-294</td>
<td>report and trace blocks</td>
<td>Record Trace 40-352</td>
</tr>
<tr>
<td>IFCID 054 - Set SQLID</td>
<td>Record Trace 40-328</td>
<td>IFCID 099 - IBM Service Record report and trace blocks</td>
</tr>
<tr>
<td>report and trace blocks</td>
<td>Record Trace 40-329</td>
<td>Record Trace 40-353</td>
</tr>
<tr>
<td>IFCID 055 - IBM Service Record report and trace blocks</td>
<td>IFCID 100 - IBM Service Record</td>
<td>Record Trace 40-354</td>
</tr>
<tr>
<td>Record Trace 40-299</td>
<td>Record Trace 40-330</td>
<td>Record Trace 40-355</td>
</tr>
<tr>
<td>IFCID 056 - IBM Service Record</td>
<td>IFCID 101 - IBM Service Record</td>
<td>Record Trace 40-356</td>
</tr>
<tr>
<td>report and trace blocks</td>
<td>report and trace blocks</td>
<td>Record Trace 40-357</td>
</tr>
<tr>
<td>Record Trace 40-298</td>
<td>IFCID 102 - IBM Service Record</td>
<td>Record Trace 40-358</td>
</tr>
<tr>
<td>IFCID 057 - IBM Service Record report and trace blocks</td>
<td>IFCID 103 - DBID/OBID Translation</td>
<td>Record Trace 40-359</td>
</tr>
<tr>
<td>Record Trace 40-299</td>
<td>report and trace blocks</td>
<td>IFCID 104 - Application Programming Defaults report and trace blocks</td>
</tr>
<tr>
<td>IFCID 058 - IBM Service Record</td>
<td>Record Trace 40-332</td>
<td>IFCID 105 - Log Data Set report and trace blocks</td>
</tr>
<tr>
<td>report and trace blocks</td>
<td>Record Trace 40-333</td>
<td>Record Trace 40-357</td>
</tr>
<tr>
<td>IFCID 059 - Fetch Start</td>
<td>Record Trace 40-334</td>
<td>IFCID 106 - Data Sharing Parameters report and trace blocks</td>
</tr>
<tr>
<td>report and trace blocks</td>
<td>Record Trace 40-335</td>
<td>Record Trace 40-361</td>
</tr>
<tr>
<td>IFCID 060 - Select Start</td>
<td>IFCID 084 - Prepare Start</td>
<td>IFCID 106 - Databases/Spaces Automatically Deferred report and trace blocks</td>
</tr>
<tr>
<td>report and trace blocks</td>
<td>report and trace blocks</td>
<td>Record Trace 40-372</td>
</tr>
<tr>
<td>Record Trace 40-305</td>
<td>Record Trace 40-336</td>
<td>IFCID 106 - Databases/Spaces Automatically Started report and trace blocks</td>
</tr>
<tr>
<td>IFCID 061 - Insert/Update/Delete Start</td>
<td>Record Trace 40-337</td>
<td>Record Trace 40-374</td>
</tr>
<tr>
<td>report and trace blocks</td>
<td>IFCID 085 - Prepare End</td>
<td>Record Trace 40-369</td>
</tr>
<tr>
<td>Record Trace 40-307</td>
<td>report and trace blocks</td>
<td>IFCID 107 - IBM Service Record report and trace blocks</td>
</tr>
<tr>
<td>IFCID 062 - DDL Start</td>
<td>Record Trace 40-338</td>
<td>Record Trace 40-376</td>
</tr>
<tr>
<td>report and trace blocks</td>
<td>IFCID 086 - Signon Start</td>
<td>Record Trace 40-377</td>
</tr>
<tr>
<td>Record Trace 40-309</td>
<td>report and trace blocks</td>
<td>Report Trace 40-374</td>
</tr>
<tr>
<td>IFCID 063 - SQL Statement</td>
<td>Record Trace 40-339</td>
<td>IFCID 107 - Distributed Data Facility Parameters report and trace blocks</td>
</tr>
<tr>
<td>report and trace blocks</td>
<td>IFCID 087 - Signon End</td>
<td>Record Trace 40-375</td>
</tr>
<tr>
<td>Record Trace 40-310</td>
<td>report and trace blocks</td>
<td>IFCID 107 - IRLM Processing Parameters report and trace blocks</td>
</tr>
<tr>
<td>IFCID 064 - Prepare Start</td>
<td>Record Trace 40-340</td>
<td>Record Trace 40-376</td>
</tr>
<tr>
<td>report and trace blocks</td>
<td>IFCID 088 - Synch Start</td>
<td>IFCID 106 - Log Initialization Parameters (Part 1) report and trace blocks</td>
</tr>
<tr>
<td>Record Trace 40-312</td>
<td>report and trace blocks</td>
<td>Record Trace 40-382</td>
</tr>
<tr>
<td>IFCID 065 - Open Cursor</td>
<td>Record Trace 40-341</td>
<td>IFCID 106 - Log Initialization Parameters (Part 2) report and trace blocks</td>
</tr>
<tr>
<td>report and trace blocks</td>
<td>IFCID 090 - DB2 Command Start</td>
<td>Record Trace 40-384</td>
</tr>
<tr>
<td>Record Trace 40-314</td>
<td>report and trace blocks</td>
<td>Report Trace 40-384</td>
</tr>
<tr>
<td>IFCID 066 - Close Cursor</td>
<td>Record Trace 40-342</td>
<td></td>
</tr>
<tr>
<td>report and trace blocks</td>
<td>Record Trace 40-343</td>
<td></td>
</tr>
<tr>
<td>Record Trace 40-317</td>
<td>Record Trace 40-344</td>
<td></td>
</tr>
<tr>
<td>IFCID 067 - Accounting</td>
<td>Record Trace 40-345</td>
<td></td>
</tr>
<tr>
<td>report and trace blocks</td>
<td>Record Trace 40-346</td>
<td></td>
</tr>
<tr>
<td>Record Trace 40-319</td>
<td>Record Trace 40-347</td>
<td></td>
</tr>
<tr>
<td>IFCID 068 - Rollback Start</td>
<td>Record Trace 40-348</td>
<td></td>
</tr>
<tr>
<td>report and trace blocks</td>
<td>Record Trace 40-349</td>
<td></td>
</tr>
<tr>
<td>Record Trace 40-320</td>
<td>Record Trace 40-351</td>
<td></td>
</tr>
<tr>
<td>IFCID 069 - IBM Service Record</td>
<td>Record Trace 40-352</td>
<td></td>
</tr>
<tr>
<td>report and trace blocks</td>
<td>Record Trace 40-353</td>
<td></td>
</tr>
<tr>
<td>Record Trace 40-321</td>
<td>Record Trace 40-354</td>
<td></td>
</tr>
<tr>
<td>IFCID 070 - Commit Phase 2 Start</td>
<td>Record Trace 40-355</td>
<td></td>
</tr>
<tr>
<td>report and trace blocks</td>
<td>Record Trace 40-356</td>
<td></td>
</tr>
<tr>
<td>Record Trace 40-322</td>
<td>Record Trace 40-357</td>
<td></td>
</tr>
<tr>
<td>IFCID 071 - IBM Service Record</td>
<td>Record Trace 40-358</td>
<td></td>
</tr>
<tr>
<td>report and trace blocks</td>
<td>Record Trace 40-359</td>
<td></td>
</tr>
<tr>
<td>Record Trace 40-323</td>
<td>Record Trace 40-361</td>
<td></td>
</tr>
<tr>
<td>IFCID 072 - Create Thread Start</td>
<td>Record Trace 40-362</td>
<td></td>
</tr>
<tr>
<td>report and trace blocks</td>
<td>Record Trace 40-363</td>
<td></td>
</tr>
<tr>
<td>Record Trace 40-324</td>
<td>Record Trace 40-364</td>
<td></td>
</tr>
</tbody>
</table>

Index X-5
<table>
<thead>
<tr>
<th>IFCID 106 - Miscellaneous Installation Parameters</th>
<th>IFCID 123 - SRV Record report and trace blocks Record Trace 40-466</th>
</tr>
</thead>
<tbody>
<tr>
<td>IFCID 106 - StoredProcedure Parameters report and trace blocks Record Trace 40-423</td>
<td>IFCID 124 - SQL Statement Record report and trace blocks Record Trace 40-467</td>
</tr>
<tr>
<td>IFCID 106 - System Initialization Parameters report and trace blocks Record Trace 40-424</td>
<td>IFCID 125 - RDBM Pool Processing report and trace blocks Record Trace 40-470</td>
</tr>
<tr>
<td>IFCID 106 - System Parameters report and trace blocks Record Trace 40-360</td>
<td>IFCID 126 - Log Buffer Write report and trace blocks Record Trace 40-472</td>
</tr>
<tr>
<td>IFCID 106 - VSAM Catalog Name Qualifier report and trace blocks Record Trace 40-443</td>
<td>IFCID 127 - Page Wait I/O In Prog (Start) report and trace blocks Record Trace 40-473</td>
</tr>
<tr>
<td>IFCID 107 - Open/Close report and trace blocks Record Trace 40-442</td>
<td>IFCID 128 - Page Wait I/O In Prog (End) report and trace blocks Record Trace 40-475</td>
</tr>
<tr>
<td>IFCID 108 - Bind Start report and trace blocks Record Trace 40-450</td>
<td>IFCID 129 - CI-S Obtained via IFI Reads report and trace blocks Record Trace 40-477</td>
</tr>
<tr>
<td>IFCID 109 - Bind End report and trace blocks Record Trace 40-448</td>
<td>IFCID 140 - Audit Auth Failures report and trace blocks Record Trace 40-478</td>
</tr>
<tr>
<td>IFCID 110 - Bind Free Start report and trace blocks Record Trace 40-449</td>
<td>IFCID 141 - Audit DDL Grant/Revoke report and trace blocks Record Trace 40-483</td>
</tr>
<tr>
<td>IFCID 111 - Bind Free End report and trace blocks Record Trace 40-454</td>
<td>IFCID 142 - Audit DDL Create/Alter/Drop report and trace blocks Record Trace 40-487</td>
</tr>
<tr>
<td>IFCID 112 - Create DDL report and trace blocks Record Trace 40-451</td>
<td>IFCID 143 - Audit First Write report and trace blocks Record Trace 40-491</td>
</tr>
<tr>
<td>IFCID 113 - Agent Allocate report and trace blocks Record Trace 40-452</td>
<td>IFCID 144 - Audit First Read report and trace blocks Record Trace 40-492</td>
</tr>
<tr>
<td>IFCID 114 - Archive Wait Start report and trace blocks Record Trace 40-457</td>
<td>IFCID 145 - Audit DML Statement report and trace blocks Record Trace 40-493</td>
</tr>
<tr>
<td>IFCID 115 - Archive Wait End DASD report and trace blocks Record Trace 40-458</td>
<td>IFCID 146 - User Record report and trace blocks Record Trace 40-498</td>
</tr>
<tr>
<td>IFCID 116 - Archive Wait End Tape report and trace blocks Record Trace 40-459</td>
<td>IFCID 147 - Data Sharing Accounting Data report and trace blocks Record Trace 40-500</td>
</tr>
<tr>
<td>IFCID 117 - Archive Read Start report and trace blocks Record Trace 40-460</td>
<td>IFCID 147 - Distributed Header Data report and trace blocks Record Trace 40-501</td>
</tr>
<tr>
<td>IFCID 118 - Archive Read End report and trace blocks Record Trace 40-461</td>
<td>IFCID 147 - Instrumentation Accounting Data report and trace blocks Record Trace 40-503</td>
</tr>
<tr>
<td>IFCID 119 - BSDS Write Start report and trace blocks Record Trace 40-462</td>
<td>IFCID 147 - Instrumentation Accounting Data Overflow report and trace blocks Record Trace 40-504</td>
</tr>
<tr>
<td>IFCID 120 - BSDS Write End report and trace blocks Record Trace 40-463</td>
<td>IFCID 147 - Logging report and trace blocks Record Trace 40-507</td>
</tr>
<tr>
<td>IFCID 121 - IBM Service Record report and trace blocks Record Trace 40-464</td>
<td>IFCID 147 - Monitor Detail Data report and trace blocks Record Trace 40-508</td>
</tr>
<tr>
<td>IFCID 122 - IBM Service Record report and trace blocks Record Trace 40-465</td>
<td>IFCID 147 - Thread Correlation Data report and trace blocks Record Trace 40-514</td>
</tr>
<tr>
<td>IFCID 147 - Thread Summary report and trace blocks Record Trace 40-499</td>
<td>IFCID 147 - Thread Summary report and trace blocks Record Trace 40-517</td>
</tr>
<tr>
<td>IFCID 148 - Resource Locking report and trace blocks Record Trace 40-517</td>
<td>IFCID 150 - Global Interest Data report and trace blocks Record Trace 40-519</td>
</tr>
<tr>
<td>IFCID 150 - Lock Resource Data report and trace blocks Record Trace 40-521</td>
<td>IFCID 150 - Lock Resource Data report and trace blocks Record Trace 40-523</td>
</tr>
<tr>
<td>IFCID 150 - Retained Lock Data report and trace blocks Record Trace 40-525</td>
<td>IFCID 150 - Suspended Lock Data report and trace blocks Record Trace 40-526</td>
</tr>
<tr>
<td>IFCID 150 - Thread Locking report and trace blocks Record Trace 40-518</td>
<td>IFCID 155 - User Record report and trace blocks Record Trace 40-530</td>
</tr>
<tr>
<td>IFCID 155 - User Record report and trace blocks Record Trace 40-531</td>
<td>IFCID 155 - User Record report and trace blocks Record Trace 40-532</td>
</tr>
<tr>
<td>IFCID 155 - User Record report and trace blocks Record Trace 40-533</td>
<td>IFCID 155 - User Record report and trace blocks Record Trace 40-534</td>
</tr>
<tr>
<td>IFCID 156 - User Record report and trace blocks Record Trace 40-535</td>
<td>IFCID 157 - DRDS RDS Interface report and trace blocks Record Trace 40-534</td>
</tr>
<tr>
<td>IFCID 158 - DRDS RDS Interface report and trace blocks Record Trace 40-535</td>
<td>IFCID 159 - DRDS Req Site Data report and trace blocks Record Trace 40-535</td>
</tr>
<tr>
<td>IFCID 160 - DC Requester report and trace blocks Record Trace 40-536</td>
<td>IFCID 160 - DC Requester report and trace blocks Record Trace 40-537</td>
</tr>
<tr>
<td>IFCID 161 - DC Server report and trace blocks Record Trace 40-538</td>
<td>IFCID 162 - DTM Request report and trace blocks Record Trace 40-539</td>
</tr>
<tr>
<td>IFCID 163 - DTM Respond report and trace blocks Record Trace 40-540</td>
<td>IFCID 164 - IBM Service Record report and trace blocks Record Trace 40-541</td>
</tr>
</tbody>
</table>
IFCID 165 - IBM Service Record report and trace blocks
Record Trace 40-542

IFCID 166 - IBM Service Record report and trace blocks
Record Trace 40-543

IFCID 167 - Conv Alloc Req Queued report and trace blocks
Record Trace 40-544

IFCID 168 - IBM Service Record report and trace blocks
Record Trace 40-545

IFCID 169 - DIST Authid Translation report and trace blocks
Record Trace 40-546

IFCID 170 - Suspend of Agent report and trace blocks
Record Trace 40-548

IFCID 171 - IBM Service Record report and trace blocks
Record Trace 40-549

IFCID 172 - Deadlock Data report and trace blocks
Record Trace 40-550

IFCID 172 - Deadlock Header report and trace blocks
Record Trace 40-551

IFCID 172 - Unit of Work - Blocker report and trace blocks
Record Trace 40-552

IFCID 172 - Unit of Work - Resource report and trace blocks
Record Trace 40-553

IFCID 173 - Class 2 Time report and trace blocks
Record Trace 40-556

IFCID 174 - Arch Log CMD Sus Start report and trace blocks
Record Trace 40-560

IFCID 174 - Arch Log CMD Sus End report and trace blocks
Record Trace 40-561

IFCID 175 - Arch Log CMD Sus End report and trace blocks
Record Trace 40-562

IFCID 177 - Package Allocation report and trace blocks
Record Trace 40-563

IFCID 178 - IBM Service Record report and trace blocks
Record Trace 40-566

IFCID 179 - IBM Service Record report and trace blocks
Record Trace 40-567

IFCID 180 - DC Communication Buffers report and trace blocks
Record Trace 40-568

IFCID 181 - IBM Service Record report and trace blocks
Record Trace 40-570

IFCID 182 - IBM Service Record report and trace blocks
Record Trace 40-571

IFCID 183 - DRDS RDS/SCC Interface report and trace blocks
Record Trace 40-572

IFCID 184 - DC Communication Buffers report and trace blocks
Record Trace 40-575

IFCID 185 - READs Data Capture Start report and trace blocks
Record Trace 40-576

IFCID 186 - IBM Service Record report and trace blocks
Record Trace 40-577

IFCID 188 - READs Data Capture End report and trace blocks
Record Trace 40-578

IFCID 190 - IBM Service Record report and trace blocks
Record Trace 40-580

IFCID 191 - 6B DSS Section report and trace blocks
Record Trace 40-582

IFCID 191 - DB2 ZEDA report and trace blocks
Record Trace 40-584

IFCID 191 - DDM Level 6B Objects report and trace blocks
Record Trace 40-588

IFCID 191 - Header Section report and trace blocks
Record Trace 40-589

IFCID 191 - DB2 ZEDA report and trace blocks
Record Trace 40-590

IFCID 191 - DDM Level 6A Header Errors report and trace blocks
Record Trace 40-591

IFCID 192 - Previous 6A Header report and trace blocks
Record Trace 40-592

IFCID 192 - Current 6A Header report and trace blocks
Record Trace 40-593

IFCID 192 - DDM Level 6A Header Errors report and trace blocks
Record Trace 40-594

IFCID 195 - SQLDA Discrepancy report and trace blocks
Record Trace 40-595

IFCID 196 - Holder report and trace blocks
Record Trace 40-596

IFCID 196 - Timeout Data report and trace blocks
Record Trace 40-597

IFCID 196 - Timeout Header report and trace blocks
Record Trace 40-601

IFCID 197 - DB2 Messages report and trace blocks
Record Trace 40-603

IFCID 198 - Buffer Manager Page Access report and trace blocks
Record Trace 40-604

IFCID 199 - Buffer Pool Statistics at Data Set Level report and trace blocks
Record Trace 40-606

IFCID 201 - Alter Buffer Pool report and trace blocks
Record Trace 40-609

IFCID 202 - Buffer Pool Attributes report and trace blocks
Record Trace 40-612

IFCID 203 - DDF Heuristic COMMIT/ROLLBK report and trace blocks
Record Trace 40-615

IFCID 204 - DDF Partner Cold Start report and trace blocks
Record Trace 40-617

IFCID 205 - As Remembered by DB2 report and trace blocks
Record Trace 40-620

IFCID 205 - As Remembered by Partner report and trace blocks
Record Trace 40-621

IFCID 205 - DDF Warm Start Log Name Error report and trace blocks
Record Trace 40-622

IFCID 206 - DDF Protocol Error report and trace blocks
Record Trace 40-623

IFCID 207 - DDF Heuristic Damage report and trace blocks
Record Trace 40-625

IFCID 208 - DDF Syncpoint Protocol Error report and trace blocks
Record Trace 40-627

IFCID 209 - DDF Syncpoint Comm Failure report and trace blocks
Record Trace 40-629

IFCID 210 - Warm Start Log Name Change report and trace blocks
Record Trace 40-631

IFCID 211 - Claim Data report and trace blocks
Record Trace 40-632

IFCID 212 - Drain Data report and trace blocks
Record Trace 40-633

IFCID 213 - Drain Lock Wait Start report and trace blocks
Record Trace 40-635

IFCID 214 - Drain Lock Wait End report and trace blocks
Record Trace 40-637

IFCID 215 - Claim Count 0 Wait Start report and trace blocks
Record Trace 40-639
IFCID 216 - Claim Count 0 Wait End
  report and trace blocks
  Record Trace 40-640
IFCID 217 - Additional Information
  report and trace blocks
  Record Trace 40-642
IFCID 217 - Agent Local Storage Pool Sizes
  report and trace blocks
  Record Trace 40-645
IFCID 217 - DBM1 Storage Pool Sizes
  report and trace blocks
  Record Trace 40-646
IFCID 217 - Storage Manager Pool Statistics
  report and trace blocks
  Record Trace 40-648
IFCID 217 - Storage Pools
  report and trace blocks
  Record Trace 40-650
IFCID 218 - Lock Avoidance Summary
  report and trace blocks
  Record Trace 40-651
IFCID 219 - Utility LISTDEF List Information
  report and trace blocks
  Record Trace 40-652
IFCID 220 - Utility Data Set Information
  report and trace blocks
  Record Trace 40-653
IFCID 221 - Buffer Pool Constrained Data (Section Type C)
  report and trace blocks
  Record Trace 40-656
IFCID 221 - Detail Buffer Pool Constrained Data (Section Type E)
  report and trace blocks
  Record Trace 40-657
IFCID 221 - Parallel Data
  report and trace blocks
  Record Trace 40-658
IFCID 221 - Parallel Group Execution
  report and trace blocks
  Record Trace 40-659
IFCID 221 - Section Type D
  report and trace blocks
  Record Trace 40-661
IFCID 222 - Parallel Group Elapsed Time
  report and trace blocks
  Record Trace 40-662
IFCID 223 - Lock Avoidance Detail
  report and trace blocks
  Record Trace 40-664
IFCID 224 - Select Procedure Bypassed
  report and trace blocks
  Record Trace 40-665
IFCID 225 - Address Space Summary - DBM1
  report and trace blocks
  Record Trace 40-667
IFCID 225 - Address Space Summary - DMBT
  report and trace blocks
  Record Trace 40-668
IFCID 225 - Shared/Common Storage Summary
  report and trace blocks
  Record Trace 40-675
IFCID 225 - Statement Cache / XPROC Detail
  report and trace blocks
  Record Trace 40-672
IFCID 225 - Storage Pool Details
  report and trace blocks
  Record Trace 40-679
IFCID 225 - System Storage Usage
  report and trace blocks
  Record Trace 40-666
IFCID 225 - Thread Information
  report and trace blocks
  Record Trace 40-681
IFCID 226 - Page Latch Contention Start
  report and trace blocks
  Record Trace 40-682
IFCID 227 - Page Latch Contention End
  report and trace blocks
  Record Trace 40-684
IFCID 228 - Archive Deallocation Start
  report and trace blocks
  Record Trace 40-685
IFCID 229 - Archive Deallocation End
  report and trace blocks
  Record Trace 40-686
IFCID 230 - Group Buffer Pool Attributes
  report and trace blocks
  Record Trace 40-687
IFCID 231 - Parallel Group Task Time
  report and trace blocks
  Record Trace 40-690
IFCID 233 - Call User Routine
  report and trace blocks
  Record Trace 40-692
IFCID 234 - Calling Agent Auth IDs
  report and trace blocks
  Record Trace 40-694
IFCID 236 - DDF SNA XLN Protocol Error
  report and trace blocks
  Record Trace 40-695
IFCID 237 - Set Current Degree
  report and trace blocks
  Record Trace 40-696
IFCID 238 - IBM Service Record
  report and trace blocks
  Record Trace 40-697
IFCID 239 - Buffer Manager Accounting Data
  report and trace blocks
  Record Trace 40-699
IFCID 239 - General Package Overflow Accounting Data
  report and trace blocks
  Record Trace 40-703
IFCID 239 - Locking Data
  report and trace blocks
  Record Trace 40-704
IFCID 239 - Overflow Package/DBRM
  report and trace blocks
  Record Trace 40-698
IFCID 239 - Package/DBRM Accounting Data
  report and trace blocks
  Record Trace 40-708
IFCID 247 - SQLDA Data and Input Host Variable Data
  report and trace blocks
  Record Trace 40-722
IFCID 248 - IBM Service Record
  report and trace blocks
  Record Trace 40-727
IFCID 249 - EDM Pool Invalidate DBD
  report and trace blocks
  Record Trace 40-726
IFCID 250 - Connect/Rebuild Group Bpool
  report and trace blocks
  Record Trace 40-732
IFCID 254 - Coupling Facility Cache Structure Statistics
  report and trace blocks
  Record Trace 40-736
IFCID 255 - Alter Group Buffer Pool
  report and trace blocks
  Record Trace 40-738
IFCID 256 - Buffer Manager PSET/Part P-Lock Request
  report and trace blocks
  Record Trace 40-740
IFCID 257 - IRLM Notify Req Detail
  report and trace blocks
  Record Trace 40-742
IFCID 258 - Data Set Extend Activity
  report and trace blocks
  Record Trace 40-743
IFCID 259 - Buffer Manager Pg P-Lock Req
  report and trace blocks
  Record Trace 40-745
IFCID 260 - IBM Service Record
  report and trace blocks
  Record Trace 40-747
IFCID 261 - Group Buffer Pool Checkpoint
  report and trace blocks
  Record Trace 40-750
IFCID 262 - GBPOOLT Castout Threshold
  processing
  report and trace blocks
  Record Trace 40-752
IFCID 263 - Page Set and Partition Castout Detail
  report and trace blocks
  Record Trace 40-752
Performance and Optimization (DSNTIP8, DSNTIP81)
report and trace blocks
System Parameters 54-61
Plan/Package Activity
report and trace blocks
Statistics 49-148
Protection Installation Parameters (DSNTIPP)
report and trace blocks
System Parameters 54-67
Protection Panel (DSNTIP11)
report and trace blocks
System Parameters 54-71

Q
Query Accelerator Preferences (DSNTIP82)
report and trace blocks
System Parameters 54-73
Query Parallelism
report and trace blocks
Accounting 5-163
Statistics 49-153
QUERY REQUESTS 34-37

R
raw EXPLAIN data
explain 13-1
Real and Auxiliary Storage for DBM1
report and trace blocks
Statistics 49-157
Real and Auxiliary Storage for DIST (DB2 10 or later)
report and trace blocks
Statistics 49-153
Real Storage in Use - Summary
report and trace blocks
Statistics 49-161
record trace 0-3
Record Trace
report and trace blocks
IFCID 001 - Checkpoint and IFI
Data 40-20
IFCID 001 - CPU Time Data 40-23
IFCID 001 - DB2 Command
Data 40-25
IFCID 001 - DDF Data by Location 40-31
IFCID 001 - Destination Related Data 40-38
IFCID 001 - Global DDF Data 40-40
IFCID 001 - IFCID Data 40-44
IFCID 001 - Log Manager Data 40-45
IFCID 001 - QSST Data 40-55
IFCID 001 - QVAS Data 40-56
IFCID 001 - QVLS Data 40-57
IFCID 001 - Subsystem Services Data 40-50
IFCID 001 - System Statistics 40-19
IFCID 001 - z/OS Metrics 40-61

Record Trace (continued)
report and trace blocks (continued)
IFCID 003 - ROWID 40-206
IFCID 003 - SQL Call Data 40-207
IFCID 004 - Trace Start 40-208
IFCID 005 - Trace Stop 40-209
IFCID 006 - Read I/O Start 40-210
IFCID 007 - Read I/O Stop 40-212
IFCID 008 - Write I/O Synch 40-213
IFCID 009 - Write I/O 40-215
IFCID 010 - Write I/O Async 40-216
IFCID 011 - Validate Exit 40-218
IFCID 012 - Edit Exit to Encode 40-219
IFCID 013 - Hash Scan Input Start 40-220
IFCID 014 - Hash Scan End 40-222
IFCID 015 - Index Scan Begin 40-223
IFCID 016 - Insert Scan Begin 40-225
IFCID 017 - Sequential Scan Begin 40-227
IFCID 018 - Scan End 40-229
IFCID 019 - Edit Exit to Decode 40-231
IFCID 020 - Lock Summary 40-232
IFCID 021 - Lock Detail 40-234
IFCID 022 - Minibind 40-239
IFCID 023 - Utility Start 40-248
IFCID 024 - Utility Change 40-252
IFCID 025 - Utility End 40-254
IFCID 026 - IBM Service Record 40-257
IFCID 027 - Sort Workfile Records 40-258
IFCID 028 - Sort Phase Detail 40-260
IFCID 029 - EDM Request Start 40-262
IFCID 030 - EDM Request End 40-264
IFCID 031 - EDM Full 40-266
IFCID 032 - Log Wait Start 40-269
IFCID 033 - IBM Service Record 40-270
IFCID 034 - Log Read Start 40-271
IFCID 035 - Log Read End 40-272
IFCID 036 - Log Non I/O Start 40-273
IFCID 037 - Log Non I/O End 40-274
IFCID 038 - Active Write Start 40-275
IFCID 039 - Active Write End 40-276
IFCID 040 - Archive Write Start 40-277
IFCID 041 - Archive Write End 40-278

Index X-11
Record Trace (continued)
report and trace blocks (continued)

IFCID 042 - Checkpoint
Start 40-279
IFCID 043 - Checkpoint
End 40-280
IFCID 044 - Lock Suspend 40-281
IFCID 045 - Lock Resume 40-285
IFCID 046 - IBM Service
Record 40-287
IFCID 047 - IBM Service
Record 40-288
IFCID 048 - IBM Service
Record 40-289
IFCID 049 - IBM Service
Record 40-290
IFCID 050 - IBM Service
Record 40-291
IFCID 051 - IBM Service
Record 40-292
IFCID 052 - IBM Service
Record 40-293
IFCID 053 - SQL
Describe/Commit/Rollback/
Remote Statement 40-294
IFCID 055 - Set SQLID 40-297
IFCID 056 - IBM Service
Record 40-298
IFCID 057 - IBM Service
Record 40-299
IFCID 058 - End SQL 40-300
IFCID 059 - Fetch Start 40-303
IFCID 060 - Select Start 40-305
IFCID 061 - Insert/Update/Delete
Start 40-307
IFCID 062 - DDL Start 40-309
IFCID 063 - SQL
Statement 40-310
IFCID 064 - Prepare Start 40-312
IFCID 065 - Open Cursor 40-314
IFCID 066 - Close Cursor 40-317
IFCID 067 - Accounting 40-319
IFCID 068 - Rollback Start 40-320
IFCID 069 - IBM Service
Record 40-321
IFCID 070 - Commit Phase 2
Start 40-322
IFCID 071 - IBM Service
Record 40-323
IFCID 072 - Create Thread
Start 40-324
IFCID 073 - Create Thread
End 40-325
IFCID 074 - Terminate Thread
Start 40-326
IFCID 075 - Terminate Thread
End 40-327
IFCID 076 - End of Memory
Start 40-328
IFCID 077 - End of Memory
End 40-329
IFCID 078 - End of Task
Start 40-330
IFCID 079 - End of Task
End 40-331
IFCID 080 - IBM Service
Record 40-332

IFCID 081 - IBM Service
Record 40-333
IFCID 082 - Identify Start 40-334
IFCID 083 - Identify End 40-335
IFCID 084 - Prepare Start 40-336
IFCID 085 - Prepare End 40-337
IFCID 086 - Signon Start 40-338
IFCID 087 - Signon End 40-339
IFCID 088 - Synch Start 40-340
IFCID 089 - Synch End 40-341
IFCID 090 - DB2 Command
Start 40-342
IFCID 091 - Command
End 40-343
IFCID 092 - AMS Command
Start 40-344
IFCID 093 - IBM Service
Record 40-345
IFCID 094 - IBM Service
Record 40-346
IFCID 095 - Sort Start 40-347
IFCID 096 - Sort End 40-348
IFCID 097 - AMS Command
End 40-351
IFCID 098 - IBM Service
Record 40-352
IFCID 099 - IBM Service
Record 40-353
IFCID 100 - IBM Service
Record 40-354
IFCID 101 - IBM Service
Record 40-355
IFCID 102 - IBM Service
Record 40-356
IFCID 103 - SOS Off 40-357
IFCID 104 - Log Data Set 40-358
IFCID 105 - DBID/OBID
Translation 40-359
IFCID 106 - Application
Programming Defaults 40-361
IFCID 107 - Data Sharing
Parameters 40-369
IFCID 108 - Databases/Spaces
Automatically Deferred 40-372
IFCID 109 - Databases/Spaces
Automatically Started 40-373
IFCID 109 - Databases/Spaces
IFCID 110 - Distributed Data
Facility Parameters 40-375
IFCID 111 - IRLM Processing
Parameters 40-379
IFCID 112 - Log Initialization
Parameters (Part 1) 40-382
IFCID 113 - Log Initialization
Parameters (Part 2) 40-384
IFCID 114 - Miscellaneous
Installation Parameters 40-389
IFCID 115 - Stored Procedures
Parameters 40-423
IFCID 116 - System Initialization
Parameters 40-424
IFCID 117 - System
Parameters 40-436

IFCID 118 - VSAM Catalog Name
Qualifier 40-437
IFCID 119 - Open/Close 40-442
IFCID 120 - Bind Start 40-443
IFCID 121 - Bind End 40-448
IFCID 122 - Bind Free
Start 40-449
IFCID 123 - Bind Free End 40-450
IFCID 124 - Thread
Allocate 40-451
IFCID 125 - Agent Allocate 40-454
IFCID 126 - Archive Wait
Start 40-457
IFCID 127 - Archive Wait End
DASD 40-458
IFCID 128 - Archive Wait End
Tape 40-459
IFCID 129 - Archive Read
Start 40-460
IFCID 130 - Archive Read
End 40-461
IFCID 131 - BDS S Write
Start 40-462
IFCID 132 - BDS S Write
End 40-463
IFCID 133 - IBM Service
Record 40-464
IFCID 134 - IBM Service
Record 40-465
IFCID 135 - SRV Record 40-466
IFCID 136 - SQL Statement
Record 40-467
IFCID 137 - RID Pool
Processing 40-470
IFCID 138 - Log Buffer
Write 40-472
IFCID 139 - Page Wait I/O In Prog
(Start) 40-473
IFCID 140 - Page Wait I/O In Prog
(End) 40-475
IFCID 141 - CI-S Obtained via IFI
Reads 40-477
IFCID 142 - Audit Auth
Failures 40-478
IFCID 143 - Audit DDL
Grant/Revoke 40-483
IFCID 144 - Audit DDL
Create/Alter/Drop 40-487
IFCID 145 - Audit First
Write 40-491
IFCID 146 - Audit First
Read 40-492
IFCID 147 - Audit DML
Statement 40-493
IFCID 148 - User Record 40-498
IFCID 149 - Data Sharing
Accounting Data 40-500
IFCID 150 - Distributed Header
Data 40-501
IFCID 151 - Instrumentation
Accounting Data 40-503
IFCID 152 - Instrumentation
Accounting Data 40-504
IFCID 153 - Logging 40-507
IFCID 225 - Address Space Summary - DIST 40-668
IFCID 225 - Shared/Common Storage Summary 40-675
IFCID 225 - Statement Cache / XPROC Detail 40-672
IFCID 225 - Storage Pool Details 40-679
IFCID 225 - System Storage Usage 40-666
IFCID 225 - Thread Information 40-681
IFCID 226 - Page Latch Contention Start 40-682
IFCID 227 - Page Latch Contention End 40-684
IFCID 228 - Archive Deallocation Start 40-685
IFCID 229 - Archive Deallocation End 40-686
IFCID 230 - Group Buffer Pool Attributes 40-687
IFCID 231 - Parallel Group Task Time 40-690
IFCID 231 - Call User Routine 40-692
IFCID 234 - Calling Agent Auth IDs 40-694
IFCID 236 - DDF SNA XLN Protocol Error 40-695
IFCID 237 - Set Current Degree 40-696
IFCID 238 - IBM Service Record 40-697
IFCID 239 - Buffer Manager Accounting Data 40-699
IFCID 239 - General Package Overflow Accounting Data 40-703
IFCID 239 - Locking Data 40-704
IFCID 239 - Overflow Package/DBRM 40-698
IFCID 239 - Package/DBRM Accounting Data 40-708
IFCID 239 - RDS Package Accounting 40-718
IFCID 239 - Resource Limit Facility 40-720
IFCID 247 - SQLDA Data and Input Host Variable Data 40-722
IFCID 248 - IBM Service Record 40-725
IFCID 249 - EDM Pool Invalidate DBD 40-726
IFCID 250 - Connect/Rebuild Connect/Disconnect Group Bpool 40-727
IFCID 251 - Buffer Manager PSET/Part P-Lock Request 40-730
IFCID 252 - IBM Service Record 40-732
IFCID 254 - Coupling Facility Cache Structure Statistics 40-733
IFCID 255 - Buffer Refresh Due to XI 40-736
IFCID 256 - Alter Group Buffer Pool 40-738
IFCID 257 - IRLM Notify Req Detail 40-740
IFCID 258 - Data Set Extend Activity 40-743
IFCID 259 - Buffer Manager Pg P-Lock Req 40-745
IFCID 260 - IBM Service Record 40-747
IFCID 261 - Group Buffer Pool Checkpoint 40-748
IFCID 262 - CBPOOLT Castout Threshold Processing 40-750
IFCID 263 - Page Set and Partition Castout Detail 40-752
IFCID 265 - IBM Service Record 40-755
IFCID 266 - IBM Service Record 40-756
IFCID 267 - CF Rebuild/Alter/Start 40-757
IFCID 268 - CF Rebuild/Alter End 40-758
IFCID 269 - Trusted/Context Trace 40-751
IFCID 270 - Trusted/Context Trace 40-764
IFCID 271 - Row Level and Column Level Access Control 40-765
IFCID 272 - Associate Locators 40-767
IFCID 273 - Allocate Cursor 40-769
IFCID 305 - Table Check Constraint 40-771
IFCID 311 - Global Temp Table Usage 40-773
IFCID 313 - Uncommitted Unit of Recovery 40-775
IFCID 314 - Authorization Exit Parameters 40-777
IFCID 315 - SQL Statement Statistics 40-779
IFCID 317 - SQL Statement String 40-787
IFCID 319 - Audit Security Record 40-788
IFCID 321 - Force-at-Commit Begin 40-790
IFCID 322 - Force-at-Commit End 40-791
IFCID 324 - Function Resolution 40-792
IFCID 325 - Trigger Activation 40-794
IFCID 329 - IXL Suspensions 40-796
IFCID 330 - Active Log Space Shortage 40-797
IFCID 331 - IBM Service Record 40-798
IFCID 332 - IBM Service Record 40-799
IFCID 333 - IBM Service Record 40-800
IFCID 335 - System Event Stalled 40-801
IFCID 342 - WF/TEMP DB Usage 40-802
IFCID 343 - MAXTEMPS Limit/Exceeded 40-804
IFCID 345 - Trace Data / SP/UDF 40-805
IFCID 346 - Package/DBRM Detail 40-806
IFCID 350 - SQL Statement 40-807
IFCID 351 - Wait TCPIP LOB 40-809
IFCID 353 - IBM Service Record 40-810
IFCID 354 - IBM Service Record 40-811
IFCID 357 - Beginning of an Index 1/O Parallel INSERT 40-812
IFCID 358 - End of an Index 1/O Parallel INSERT 40-813
IFCID 359 - Index Page Split 40-814
IFCID 361 - Audit Admin Authorities 40-815
IFCID 362 - Start Trace and Stop Trace with Audit Policy 40-818
IFCID 363 - Data Section QW0363 40-821
IFCID 363 - Data Section QW0363E 40-823
IFCID 363 - Parallel Straw Model Performance Trace 40-820
IFCID 365 - Remote Location Statistics 40-825
IFCID 366 - Incompatible Function Char(DEC) Executed 40-828
IFCID 369 - Aggregated Accounting Statistics 40-830
IFCID 370 - Database Open Information 40-833
IFCID 371 - Database Close Information 40-835
IFCID 377 - Pseudo Delete Daemon Cleanup 40-837
IFCID 378 - Accel. Call Event Begin (DB2 10 or later) 40-838
IFCID 379 - Accel. Call Event End (DB2 10 or later) 40-839
IFCID 380 - Stored Procedure Detail Record 40-840
IFCID 381 - UDF Detail Record 40-843
IFCID 384 - IBM Service Record 40-846
IFCID 385 - IBM Service Record 40-847
IFCID 386 - IBM Service Record 40-848
Record Trace (continued)

report and trace blocks (continued)

IFCID 397 - IBM Service
- Record 40-549
- IFCID 398 - IBM Service
- Record 40-580
- IFCID 399 - IBM Service
- Record 40-851
- IFCID 401 - Static Statements in EDM Pool
- Record 40-852
- IFCID 402 - System Profile - Monitoring Statistics (DB2 10 or later)
- Record 40-856
- IFCID 407 - Non Nested Statement ID Record
- Record 40-858
- IFCID 498 - UDF Statement ID Record
- Record 40-859
- IFCID 499 - Stored Procedure
- Statement ID Record 40-860
- report
- audit summary 8-1
- lock detail report 33-15
- Lock Suspension 33-2
- lockout 33-12
- SQL activity 44-1
- report and trace blocks
- Accelerator
- Accounting 5-48
- Statistics 49-3
- Accounting Rollup
- Statistics 49-9
- Aggregated Accounting Statistics
- Statistics 49-10
- Alter Buffer Pool Command
- Issued System Parameters 55-1
- Alter Group Buffer Pool Command
- Issued System Parameters 56-1
- Application Programming Defaults
- Panel 1 (DSNTIPF)
- System Parameters 54-3
- Application Programming Defaults
- Panel 2 (DSNTIP4, DSNTIP41)
- System Parameters 54-9
- Archive Log Installation Parameters (DSNTIPA)
- System Parameters 54-13
- Authorization Management
- Statistics 49-12
- Buffer Pool Activity
- Accounting 5-52
- Buffer Pool General
- Statistics 49-15
- Buffer Pool Parameters
- System Parameters 57-1
- Buffer Pool Parameters (DSNTIP1)
- System Parameters 54-17
- Buffer Pool Read
- Statistics 49-20
- Buffer Pool Sort/Merge
- Statistics 49-28
- Buffer Pool Write
- Statistics 49-31
- Common Storage Below and Above 2 GB (DB2 10 or later)
- Statistics 49-97
- CPU and Storage Metrics
- Statistics 49-39
- CPU Times
- Statistics 49-41
- Data Capture
- Accounting 5-56
- Statistics 49-44
- Data Definition Control Support (DSNTIPZ)
- System Parameters 54-19
- Data Set Statistics
- Statistics 49-45
- Data Sharing Locking
- Accounting 5-58
- Statistics 49-47
- Databases and Spaces Started
- Automatically (DSNTIPS)
- System Parameters 54-24
- DB2 API
- Statistics 49-68
- DB2 Commands
- Statistics 49-69
- DB2 Utilities Parameters (DSNTIP6, DSNTIP61, DSNTIP62)
- System Parameters 54-32
- DB2 Version Install (DSNTIPA1)
- System Parameters 54-36
- DBM1 and MVS Storage Below 2 GB
- Statistics 49-52
- DBM1 Storage Above 2 GB
- Statistics 49-59
- Default Startup Modules (DSNTIPO3)
- System Parameters 54-25
- Define Group or Member (DSNTIPK)
- System Parameters 54-22
- DIST and MVS Storage Below 2 GB (DB2 10 or later)
- Statistics 49-77
- DIST Storage Above 2 GB (DB2 10 or later)
- Statistics 49-76
- Distributed Activity - Requester (Short Report)
- Accounting 5-18
- Distributed Activity Requester (Short Trace)
- Accounting 5-33
- Distributed Activity Server (Short Report)
- Accounting 5-15
- Distributed Activity Server (Short Trace)
- Accounting 5-30
- Distributed Data Facility Panel 1 (DSNTIPR)
- System Parameters 54-26
- Distributed Data Facility Panel 2 (DSNTIPS)
- System Parameters 54-29
- Drain and Claim
- Accounting 5-83
- DRDA Remote Locations
- Statistics 49-80
- Dynamic SQL Statement
- Accounting 5-84
- Statistics 49-66
- EDM Pool Activity
- Statistics 49-86
- EDM Pool Activity (DB2 10 or later)
- Statistics 49-93
- EDM Pool Activity (DB2 9)
- Statistics 49-97
- General (Short Report)
- Accounting 5-7
- General (Short Trace)
- Accounting 5-21
- Global Contention L-Locks
- Accounting 5-86
- Global Contention P-Locks
- Accounting 5-88
- Global DDF Activity
- Statistics 49-98
- Group Buffer Pool Activity
- Accounting 5-90
- Statistics 49-103
- Group Buffer Pool Parameters
- System Parameters 58-1
- Highlights
- Accounting 5-94
- Statistics 49-115
- Highlights - Report
- Accounting 5-95
- Highlights - Trace
- Accounting 5-101
- Identification
- Accounting 5-106
- IFC Destinations
- Statistics 49-121
- IFC Record Counts
- Statistics 49-125
- IFCID 001 - Checkpoint and IFI Data
- Record Trace 40-20
- IFCID 001 - CPU Time Data
- Record Trace 40-23
- IFCID 001 - DB2 Command Data
- Record Trace 40-25
- IFCID 001 - DDF Data by Location
- Record Trace 40-31
- IFCID 001 - Destination Related Data
- Record Trace 40-38
- IFCID 001 - Global DDF Data
- Record Trace 40-40
- IFCID 001 - IFCID Data
- Record Trace 40-44
- IFCID 001 - Log Manager Data
- Record Trace 40-45
- IFCID 001 - QSST Data
- Record Trace 40-55
- IFCID 001 - QVLS Data
- Record Trace 40-57
- IFCID 001 - Subsystem Services Data
- Record Trace 40-50
- IFCID 001 - System Statistics
- Record Trace 40-19
IFCID 001 - z/OS Metrics
  Record Trace 40-61
IFCID 002 - Accelerator Data
  Record Trace 40-64
IFCID 002 - Buffer Pool Activity
  Record Trace 40-69
IFCID 002 - Data Manager Data
  Record Trace 40-85
IFCID 002 - Data Sharing Locking Data
  Record Trace 40-92
IFCID 002 - Dynamic SQL Statement
  Record Trace 40-63
IFCID 002 - EDM Pool Data
  Record Trace 40-95
IFCID 002 - Group Buffer Pools Activity Data
  Record Trace 40-103
IFCID 002 - Locking Data
  Record Trace 40-112
IFCID 002 - Miscellaneous
  Record Trace 40-116
IFCID 002 - Nested SQL Activity
  Record Trace 40-118
IFCID 002 - Query Parallelism
  Record Trace 40-120
IFCID 002 - RID List Processing
  Record Trace 40-124
IFCID 002 - ROWID
  Record Trace 40-126
IFCID 002 - Service Controller Data
  Record Trace 40-127
IFCID 002 - SQL Call Data
  Record Trace 40-135
IFCID 003 - Accelerator Data
  Record Trace 40-146
IFCID 003 - Accounting
  Record Trace 40-144
IFCID 003 - Buffer Manager Accounting Data
  Record Trace 40-149
IFCID 003 - Data Sharing Accounting Data
  Record Trace 40-153
IFCID 003 - Data Sharing Locking Data
  Record Trace 40-154
IFCID 003 - DDF Data by Location
  Record Trace 40-156
IFCID 003 - Dynamic SQL Statement
  Record Trace 40-165
IFCID 003 - Group Buffer Pools Activity Data
  Record Trace 40-167
IFCID 003 - IFI Class 5 Times and Data Capture
  Record Trace 40-169
IFCID 003 - Initial DB2 Requester and MVS Correlation Data
  Record Trace 40-171
IFCID 003 - Instrumentation Accounting Data
  Record Trace 40-175

Record Trace 40-175

Record Trace 40-177

Record Trace 40-179

Record Trace 40-181

Record Trace 40-183

Record Trace 40-185

Record Trace 40-187

Record Trace 40-189

Record Trace 40-191

Record Trace 40-193

Record Trace 40-195

Record Trace 40-197

Record Trace 40-199

Record Trace 40-201

Record Trace 40-203

Record Trace 40-205

Record Trace 40-207

Record Trace 40-209

Record Trace 40-211

Record Trace 40-213

Record Trace 40-215

Record Trace 40-217

Record Trace 40-219

Record Trace 40-221

Record Trace 40-223

Record Trace 40-225

Record Trace 40-227

Record Trace 40-229

Record Trace 40-231

Record Trace 40-233

Record Trace 40-235

Record Trace 40-237

Record Trace 40-239

Record Trace 40-241

Record Trace 40-243

Record Trace 40-245

Record Trace 40-247

Record Trace 40-249

Record Trace 40-251

Record Trace 40-253

Record Trace 40-255

Record Trace 40-257

Record Trace 40-259

Record Trace 40-261

Record Trace 40-263

Record Trace 40-265

Record Trace 40-267

Record Trace 40-269

Record Trace 40-271

Record Trace 40-273

Record Trace 40-275

Record Trace 40-277

Record Trace 40-279

Record Trace 40-281

Record Trace 40-283

Record Trace 40-285

Record Trace 40-287

Record Trace 40-289

Record Trace 40-291

Record Trace 40-293

Record Trace 40-295

Record Trace 40-297

Record Trace 40-299
report and trace blocks (continued)
IFCID 144 - Audit First Read
Record Trace 40-492
IFCID 145 - Audit DML Statement
Record Trace 40-493
IFCID 146 - User Record
Record Trace 40-498
IFCID 147 - Data Sharing Accounting Data
Record Trace 40-500
IFCID 147 - Distributed Header Data
Record Trace 40-501
IFCID 147 - Instrumentation Accounting Data
Record Trace 40-503
IFCID 147 - Instrumentation Accounting Data Overflow
Record Trace 40-504
IFCID 147 - Logging
Record Trace 40-507
IFCID 147 - Monitor Detail Data
Record Trace 40-508
IFCID 147 - Thread Correlation Data
Record Trace 40-514
IFCID 147 - Thread Summary
Record Trace 40-499
IFCID 149 - Resource Locking
Record Trace 40-517
IFCID 150 - Global Interest Data
Record Trace 40-519
IFCID 150 - Held Lock Data
Record Trace 40-521
IFCID 150 - Lock Resource Data
Record Trace 40-523
IFCID 150 - Retained Lock Data
Record Trace 40-525
IFCID 150 - Suspend Lock Data
Record Trace 40-526
IFCID 150 - Suspend Locking
Record Trace 40-518
IFCID 151 - User Record
Record Trace 40-528
IFCID 152 - User Record
Record Trace 40-529
IFCID 153 - User Record
Record Trace 40-530
IFCID 154 - User Record
Record Trace 40-531
IFCID 155 - User Record
Record Trace 40-532
IFCID 156 - User Record
Record Trace 40-533
IFCID 157 - DRDS RDS Interface
Record Trace 40-534
IFCID 158 - DRDS CNV Interface
Record Trace 40-535
IFCID 159 - DRDS Req Site Data
Record Trace 40-536
IFCID 160 - DC Requester
Record Trace 40-537
IFCID 161 - DC Server
Record Trace 40-538
IFCID 162 - DTM Request
Record Trace 40-539
IFCID 163 - DTM Respond
Record Trace 40-540
report and trace blocks (continued)
IFCID 164 - IBM Service Record
Record Trace 40-541
IFCID 165 - IBM Service Record
Record Trace 40-542
IFCID 166 - IBM Service Record
Record Trace 40-543
IFCID 167 - Conv Alloc Req Queued
Record Trace 40-544
IFCID 168 - IBM Service Record
Record Trace 40-545
IFCID 169 - DIST Authid Translation
Record Trace 40-546
IFCID 170 - Suspend of Agent
Record Trace 40-548
IFCID 171 - IBM Service Record
Record Trace 40-549
IFCID 172 - Deadlock Data
Record Trace 40-550
IFCID 172 - Deadlock Header
Record Trace 40-551
IFCID 172 - Unit of Work - Blocker
Record Trace 40-553
IFCID 172 - Unit of Work - Resource
Record Trace 40-552
IFCID 172 - Unit of Work - Waiter
Record Trace 40-556
IFCID 173 - Class 2 Time
Record Trace 40-560
IFCID 174 - Arch Log CMD Sus Start
Record Trace 40-561
IFCID 175 - Arch Log CMD Sus End
Record Trace 40-562
IFCID 177 - Package Allocation
Record Trace 40-563
IFCID 178 - IBM Service Record
Record Trace 40-566
IFCID 179 - IBM Service Record
Record Trace 40-567
IFCID 180 - DC Communication Buffers
Record Trace 40-568
IFCID 181 - IBM Service Record
Record Trace 40-570
IFCID 182 - IBM Service Record
Record Trace 40-571
IFCID 183 - DRDS RDS/SCC Interface
Record Trace 40-572
IFCID 184 - DC Communication Buffers
Record Trace 40-575
IFCID 185 - READs Data Capture Start
Record Trace 40-576
IFCID 186 - IBM Service Record
Record Trace 40-577
IFCID 188 - READs Data Capture End
Record Trace 40-578
IFCID 190 - IBM Service Record
Record Trace 40-580
IFCID 191 - IBDR Section
Record Trace 40-588
IFCID 191 - Command and/or Reply Section
Record Trace 40-582
IFCID 191 - DB2 ZEDA
Record Trace 40-584
report and trace blocks (continued)
IFCID 191 - DB2 ZEDA
Record Trace 40-584
IFCID 191 - DB2 ZEDA
Record Trace 40-584
IFCID 191 - Late Descriptor Section
Record Trace 40-587
IFCID 192 - Current 6A Header
Record Trace 40-590
IFCID 192 - DDM Level 6A Header Errors
Record Trace 40-589, 40-591
IFCID 192 - Previous 6A Header
Record Trace 40-592
IFCID 193 - UOW/SQLCODE Mismatch
Record Trace 40-593
IFCID 194 - Invalid SNA FMH-5
Record Trace 40-594
IFCID 195 - SQLDA Discrepancy
Record Trace 40-595
IFCID 196 - Holder
Record Trace 40-598
IFCID 196 - Timeout Data
Record Trace 40-597
IFCID 196 - Timeout Header
Record Trace 40-601
IFCID 197 - DB2 Messages
Record Trace 40-605
IFCID 198 - Buffer Manager Page Access
Record Trace 40-604
IFCID 199 - Buffer Pool Statistics at Data Set Level
Record Trace 40-606
IFCID 201 - Alter Buffer Pool
Record Trace 40-609
IFCID 202 - Buffer Pool Attributes
Record Trace 40-612
IFCID 203 - DDF Heuristic COMMIT/ROLLBK
Record Trace 40-615
IFCID 204 - DDF Partner Cold Start
Record Trace 40-617
IFCID 205 - As Remembered by DB2
Record Trace 40-620
IFCID 205 - As Remembered by Partner
Record Trace 40-621
IFCID 205 - DDF Warm Start Log Name Error
Record Trace 40-622
IFCID 205 - DDF Warm Start Log Name Error information
Record Trace 40-619
IFCID 206 - DDF Protocol Error
Record Trace 40-623
IFCID 207 - DDF Heuristic Damage
Record Trace 40-625
IFCID 208 - DDF Syncpoint Protocol Error
Record Trace 40-627
IFCID 209 - DDF Syncpoint Comm Failure
Record Trace 40-629
REPORT AND TRACE BLOCKS (CONTINUED)

IFCID 210 - Warm Start Log Name Change
  Record Trace 40-631
IFCID 211 - Claim Data
  Record Trace 40-632
IFCID 212 - Drain Data
  Record Trace 40-633
IFCID 213 - Drain Lock Wait Start
  Record Trace 40-635
IFCID 214 - Drain Lock Wait End
  Record Trace 40-637
IFCID 215 - Claim Count 0 Wait Start
  Record Trace 40-639
IFCID 216 - Claim Count 0 Wait End
  Record Trace 40-640
IFCID 217 - Additional Information
  Record Trace 40-642
IFCID 217 - Agent Local Storage Pool Sizes
  Record Trace 40-643
IFCID 217 - DBM1 Storage Pool Sizes
  Record Trace 40-646
IFCID 217 - Storage Manager Pool Statistics
  Record Trace 40-648
IFCID 217 - Storage Pools
  Record Trace 40-641
IFCID 218 - Lock Avoidance Summary
  Record Trace 40-651
IFCID 219 - Utility LISTDEF List Information
  Record Trace 40-652
IFCID 220 - Utility Data Set Information
  Record Trace 40-653
IFCID 221 - Buffer Pool Constrained Data (Section Type C)
  Record Trace 40-656
IFCID 221 - Detail Buffer Pool Constrained Data (Section Type E)
  Record Trace 40-657
IFCID 221 - Parallel Data
  Record Trace 40-658
IFCID 221 - Parallel Group Execution
  Record Trace 40-655
IFCID 221 - Section Type D
  Record Trace 40-661
IFCID 222 - Parallel Group Elapsed Time
  Record Trace 40-662
IFCID 222 - Lock Avoidance Detail
  Record Trace 40-664
IFCID 222 - Select Procedure Bypassed
  Record Trace 40-665
IFCID 225 - Address Space Summary - DBM1
  Record Trace 40-667
IFCID 225 - Address Space Summary - DIST
  Record Trace 40-668
IFCID 225 - Shared/Common Storage Summary
  Record Trace 40-675
IFCID 225 - Statement Cache
  Record Trace 40-672
IFCID 225 - Storage Pool Details
  Record Trace 40-679
IFCID 225 - System Storage Usage
  Record Trace 40-666
IFCID 225 - Thread Information
  Record Trace 40-681
IFCID 226 - Page Latch Contention Start
  Record Trace 40-682
IFCID 227 - Page Latch Contention End
  Record Trace 40-684
IFCID 228 - Archive Deallocation Start
  Record Trace 40-685
IFCID 229 - Archive Deallocation End
  Record Trace 40-686
IFCID 230 - Group Buffer Pool Attributes
  Record Trace 40-687
IFCID 231 - Parallel Group Task Time
  Record Trace 40-690
IFCID 232 - Call User Routine
  Record Trace 40-692
IFCID 234 - Calling Agent Auth IDs
  Record Trace 40-694
IFCID 236 - DDF SNA XLN Protocol Error
  Record Trace 40-695
IFCID 237 - Set Current Degree
  Record Trace 40-696
IFCID 238 - IBM Service Record
  Record Trace 40-697
IFCID 239 - Buffer Manager Accounting Data
  Record Trace 40-699
IFCID 239 - General Package Accounting Data
  Record Trace 40-703
IFCID 239 - Locking Data
  Record Trace 40-704
IFCID 239 - Overflow Package/DBRM Accounting Data
  Record Trace 40-698
IFCID 240 - Package/DBRM Accounting Data
  Record Trace 40-707
IFCID 240 - RDS Package Accounting
  Record Trace 40-718
IFCID 240 - Resource Limit Facility
  Record Trace 40-720
IFCID 247 - SQLDA Data and Input Host Variable Data
  Record Trace 40-722
IFCID 248 - IBM Service Record
  Record Trace 40-725
IFCID 249 - EDM Pool Invalidate DBD
  Record Trace 40-726
IFCID 250 - Connect/Rebuild Connect/Disconnect Group Bpool
  Record Trace 40-727
IFCID 251 - Buffer Manager PSET/Part P-Lock Request
  Record Trace 40-730
IFCID 252 - IBM Service Record
  Record Trace 40-732

INDEX X-19
report and trace blocks (continued)
IFCID 325 - Trigger Activation
  Record Trace 40-794
IFCID 329 - IXL Suspensions
  Record Trace 40-796
IFCID 330 - Active Log Space Shortage
  Record Trace 40-797
IFCID 331 - IBM Service Record
  Record Trace 40-798
IFCID 332 - IBM Service Record
  Record Trace 40-799
IFCID 333 - IBM Service Record
  Record Trace 40-800
IFCID 335 - System Event Stalled
  Record Trace 40-801
IFCID 342 - WF/TEMP DB Usage
  Record Trace 40-802
IFCID 343 - MAXTEMPS Limit/Exceeded
  Record Trace 40-804
IFCID 345 - Trace Data / SP/UDF
  Record Trace 40-805
IFCID 346 - Package/DBRM Detail
  Record Trace 40-806
IFCID 349 - SQL Statement
  Record Trace 40-807
IFCID 350 - SQL Statement
  Record Trace 40-808
IFCID 351 - Wait TCP/IP LOB
  Record Trace 40-809
IFCID 353 - IBM Service Record
  Record Trace 40-810
IFCID 354 - IBM Service Record
  Record Trace 40-811
IFCID 357 - Beginning of an Index
  I/O Parallel INSERT
  Record Trace 40-812
IFCID 358 - End of an Index I/O
  Parallel INSERT
  Record Trace 40-813
IFCID 359 - Index Page Split
  Record Trace 40-814
IFCID 361 - Audit Admin Authorities
  Record Trace 40-815
IFCID 362 - Start Trace and Stop Trace
  with Audit Policy
  Record Trace 40-816
IFCID 363 - Data Section QW0363
  Record Trace 40-821
IFCID 363 - Data Section QW0363E
  Record Trace 40-823
IFCID 365 - Parallel Straw Model
  Performance Trace
  Record Trace 40-820
IFCID 365 - Remote Location Statistics
  Record Trace 40-825
IFCID 366 - Incompatible Function
  Char(DEC) Executed
  Record Trace 40-828
IFCID 369 - Aggregated Accounting
  Statistics
  Record Trace 40-830
IFCID 370 - Database Open Information
  Record Trace 40-833
IFCID 371 - Database Close Information
  Record Trace 40-835
report and trace blocks (continued)
IFCID 377 - Pseudo Delete Daemon Cleanup
  Record Trace 40-837
IFCID 378 - Accel. Call Event Begin
  (DB2 10 or later)
  Record Trace 40-838
IFCID 379 - Accel. Call Event End
  (DB2 10 or later)
  Record Trace 40-839
IFCID 380 - Stored Procedure Detail
  Record
  Record Trace 40-840
IFCID 381 - UDF Detail Record
  Record Trace 40-843
IFCID 383 - IBM Service Record
  Record Trace 40-846
IFCID 385 - IBM Service Record
  Record Trace 40-850
IFCID 388 - IBM Service Record
  Record Trace 40-851
IFCID 399 - IBM Service Record
  Record Trace 40-851
IFCID 401 - Static Statements in EDM Pool
  Record Trace 40-852
IFCID 402 - System Profile - Monitoring Statistics (DB2 10 or later)
  Record Trace 40-856
IFCID 407 - Non Nested Statement ID Record
  Record Trace 40-858
IFCID 408 - UDF Statement ID Record
  Record Trace 40-859
IFCID 499 - Stored Procedure Detail
  Record
  Record Trace 40-860
Initial DB2 Common Server Correlation
  Accounting 5-113
  Initial DB2 Common Server or Universal JDBC Driver Correlation
  Accounting 5-115
Initial DB2 Requester Correlation
  Accounting 5-117
Initial Other Requester Correlation
  Accounting 5-120
IRLM Installation Parameters (DSNTIPI)
  System Parameters 54-37
IRLM Processing Parameters
  System Parameters 54-40
IRLM Storage Below and Above 2 GB (DB2 11)
  Statistics 49-127
Latch Counters
  Statistics 49-129
List of Long Names
  System Parameters 54-49
Lock Escalation Parameters (DSNTP8)
  System Parameters 54-42
report and trace blocks (continued)
Locking
  Accounting 5-122
  Locking Activity
  Statistics 49-133
Log Activity
  Statistics 49-137
Log Installation Parameters
  (DSNTIPL, DSNTIP81)
  System Parameters 54-43
Logging Activity
  Accounting 5-125
Miscellaneous
  Accounting 5-126
  Statistics 49-142
MVS Accounting
  Accounting 5-127
MVS LPAR Shared Storage Above 2 GB (DB2 10 or later)
  Statistics 49-144
MVS Parmib Update Parameters (DSNTPM)
  System Parameters 54-50
Open/Close Activity
  Statistics 49-146
Operator Functions Installation Parameters (DSNTP8)
  System Parameters 54-51
Other System Parameters
  System Parameters 54-55
Package Buffer Pool Activity - Class 10
  Accounting 5-128
Package General (Short Report)
  Accounting 5-12
Package General (Short Trace)
  Accounting 5-28
Package Global Contention L-Locks - Class 8
  Accounting 5-132
Package Global Contention P-Locks - Class 8
  Accounting 5-134
Package Identification
  Accounting 5-136
Package Identification - Report
  Accounting 5-137
Package Identification - Trace
  Accounting 5-142
Package Locking Activity - Class 10
  Accounting 5-147
Package SQL Activity - Class 10
  Accounting 5-150
Package Times - Class 7
  Accounting 5-160
Package Times - Class 8 - Suspensions
  Accounting 5-152
Performance and Optimization
  (DSNTP8, DSNTP81)
  System Parameters 54-61
Plan/Package Activity
  Statistics 49-148
Protection Installation Parameters (DSNTPM)
  System Parameters 54-67
Protection Panel (DSNTP81)
  System Parameters 54-71
Statistics (continued)
report and trace blocks (continued)
DBM1 and MVS Storage Below 2 GB 49-52
DBM1 Storage Above 2 GB 49-59
DIST and MVS Storage Below 2 GB (DB2 10 or later) 49-77
DIST Storage Above 2 GB (DB2 10 or later) 49-76
DRDA Remote Locations 49-80
Dynamic SQL Statement 49-65
EDM Pool Activity 49-86
EDM Pool Activity (DB2 10 or later) 49-93
EDM Pool Activity (DB2 9) 49-87
Global DDF Activity 49-98
Group Buffer Pool Activity 49-103
Highlights 49-115
IFC Destinations 49-121
IFC Record Counts 49-125
IRLM Storage Below and Above 2 GB (DB2 11) 49-127
Latch Counters 49-129
Locking Activity 49-133
Log Activity 49-137
Miscellaneous 49-142
MVS LPAR Shared Storage Above 2 GB (DB2 10 or later) 49-144
Open/Close Activity 49-146
Plan/Package Activity 49-148
Query Parallelism 49-153
Real and Auxiliary Storage for DBM1 49-157
Real and Auxiliary Storage for DIST (DB2 10 or later) 49-159
Real Storage in Use - Summary 49-161
RID List Processing 49-162
ROWID 49-166
Short-on-Storage Metrics 49-167
SQL DCL 49-177
SQL DDL 49-179
SQL DML 49-185
Stored Procedures 49-168
Subsystem Services 49-169
Subsystem Shared Storage Above 2 GB (DB2 10 or later) 49-175
Triggers 49-187
Use Currently Committed 49-188
User-Defined Functions 49-189
Workfile Database 49-190
Statistics report 0-49
Storage Sizes Installation Parms (DSNTIPC, DSNTIPE, DSNTIPE1)
report and trace blocks
System Parameters 54-82
Stored Procedures report and trace blocks
Accounting 5-178
Statistics 49-168
Subsystem Services report and trace blocks
Statistics 49-169
Subsystem Shared Storage Above 2 GB (DB2 10 or later) report and trace blocks
Statistics 49-175

Summary
SQL activity 43-2
summary report explain 22-1
support home website xxix
syntax diagrams xxvi
system parameters 0-3
example 53-1
header 52-1

System Parameters
report and trace blocks
Alter Buffer Pool Command
Issued 55-1
Alter Group Buffer Pool Command
Issued 56-1
Application Programming Defaults Panel 1 (DSNTIPF) 54-3
Application Programming Defaults Panel 2 (DSNTIP4, DSNTIP41) 54-9
Archive Log Installation Parameters (DSNTIPA) 54-13
Buffer Pool Parameters
Buffer Pool Parameters (DSNTIP1) 54-17
Data Definition Control Support (DSNTIFZ) 54-19
Databases and Spaces Started Automatically (DSNTIPS) 54-24
DB2 Utilities Parameters (DSNTIP6, DSNTIP61, DSNTIP62) 54-32
DB2 Version Install (DSNTIP1A) 54-36
Default Startup Modules (DSNTIP3) 54-25
Define Group or Member (DSNTIFK) 54-22
Distributed Data Facility Panel 1 (DSNTIFR) 54-26
Distributed Data Facility Panel 2 (DSNTIPS) 54-29
Group Buffer Pool Parameters 58-1
IRLM Installation Parameters (DSNTIP1) 54-37
IRLM Processing Parameters 54-40
List of Long Names 54-49
Lock Escalation Parameters (DSNTIPF) 54-42
Log Installation Parameters (DSNTIPL, DSNTIP41) 54-43
MVS Parmlib Update Parameters (DSNTIPM) 54-50
Operator Functions Installation Parameters (DSNTIPO) 54-51
Other System Parameters 54-55
Performance and Optimization (DSNTIP8, DSNTIP81) 54-61
Protection Installation Parameters (DSNTIP) 54-67
Protection Panel (DSNTIP1) 54-71
Query Accelerator Preferences (DSNTIP2) 54-73

System Parameters (continued)
report and trace blocks (continued)
Routine Parameters (DSNTIPX) 54-75
Sizes Panel 1 (DSNTIPD) 54-77
SQL Object Defaults Panel (DSNTIP7, DSNTIP71) 54-78
Storage Sizes Installation Parms (DSNTIPC, DSNTIPE, DSNTIPE1) 54-82
Tracing, Checkpoint & Pseudo-Close Parameters (DSNTIPN) 54-88
Workfile Database Panel (DSNTIP1) 54-92

Table and table space data explain 18-1
Termination - Abnormal report and trace blocks
Accounting 5-191
Termination - In Doubt report and trace blocks
Accounting 5-192
Termination - Normal report and trace blocks
Accounting 5-193
terminology online xxviii
terminology used xxv
Times - Class 1 - Application Time
report and trace blocks
Accounting 5-195
Times - Class 1 - Elapsed Time Distribution
report and trace blocks
Accounting 5-201
Times - Class 2 - DB2 Time
report and trace blocks
Accounting 5-202
Times - Class 2 - Time Distribution report and trace blocks
Accounting 5-209
Times - Class 3 - Suspensions report and trace blocks
Accounting 5-210
Times - Class 5 - IFI Time report and trace blocks
Accounting 5-219
Times - Class 7 - CP CPU Distribution report and trace blocks
Accounting 5-221
Times - Class 7 - Elapsed Time Distribution
report and trace blocks
Accounting 5-222

trace
audit 9-1
lock detail 34-30
locking 34-1
SQL activity 45-1
trace blocks
Initial DB2 Common Server Accounting 5-112
Tracing, Checkpoint & Pseudo-Close
Parameters (DSNTIPN)
  report and trace blocks
System Parameters  54-88
Triggers
  report and trace blocks
Accounting  5-224
Statistics  49-187
Truncated Values
  report and trace blocks
Accounting  5-225

U
updates  xxix
Use Currently Committed
  report and trace blocks
Statistics  49-188
User-Defined Functions
  report and trace blocks
Accounting  5-226
Statistics  49-189
utility access detail (Type UTILITY)
  audit  9-35
utility access summary (UTILITY)
  audit  8-18

V
VSAM data sets  64-1

W
where to find information  xxviii
Workfile Database
  report and trace blocks
Statistics  49-190
Workfile Database Panel (DSNTIP91)
  report and trace blocks
System Parameters  54-92
workload detail
  SQL activity  43-5
Workload Detail
  Lock Suspension Activity  62-12
Readers’ Comments — We’d Like to Hear from You

IBM Tivoli OMEGAMON XE for DB2 Performance Expert on z/OS
IBM Tivoli OMEGAMON XE for DB2 Performance Monitor on z/OS
Report Reference
Version 5.2.0

Publication No. SH12-6991-00

We appreciate your comments about this publication. Please comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. The comments you send should pertain to only the information in this manual or product and the way in which the information is presented.

For technical questions and information about products and prices, please contact your IBM branch office, your IBM business partner, or your authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you. IBM or any other organizations will only use the personal information that you supply to contact you about the issues that you state on this form.

Comments:

Thank you for your support.

Send your comments to the address on the reverse side of this form.

If you would like a response from IBM, please fill in the following information:

Name

Address

Company or Organization

Phone No.

Email address