IBM IMS High Performance Load for z/OS
Version 2 Release 1

User's Guide
IBM IMS High Performance Load for z/OS
Version 2 Release 1

User's Guide
Contents

About this information ........................................ vii

Part 1. IMS High Performance Load overview .................. 1

Chapter 1. Introduction to IMS High Performance Load ........... 3
What’s new in IMS High Performance Load .................. 4
What does IMS High Performance Load do? ............... 9
Features and benefits ...................................... 12
Restrictions ............................................. 13
What’s new in Version 2 ................................ 14
Integration with other utilities and products ............ 15
Compatibility with former products ...................... 16
  Compatibility with IMS High Performance Load
    for OS/390 Version 1 .................................. 16
  Compatibility with IMS DBT Version 2 .................. 17
Utilities management solutions .......................... 18
Service updates and support information ............... 19
IMS HP Load documentation and updates .............. 20
Accessibility features for IMS HP Load ................ 21

Chapter 2. Installing and configuring IMS High Performance Load ........ 23
Hardware and software prerequisites .................. 24
Configuring the IMS Tools Knowledge Base server .... 25
Configuring for DBRC command authorization ....... 27
Configuring the environment to use the Load API .... 28

Part 2. IMS High Performance Load .......................... 29

Chapter 3. Reloading IMS databases with IMS High Performance Load .......... 31
Data flow ............................................. 32
Restrictions ........................................ 33
Reloading a database .................................. 34
JCL requirements ..................................... 35
Control statements ..................................... 45
  Overview of control statements ...................... 45
  BMSETDSG control statement ....................... 49
  BUFFERNO control statement ..................... 50
  CHKEMPTYUL control statement .................... 50
  CHKTRAIL control statement ....................... 51
  CHKUPDT control statement ....................... 52
  COMPAUTH control statement ...................... 52
  COMPRESS control statement ...................... 53
  DATXEXIT control statement ...................... 54
  DBDSNAP control statement ....................... 55
  DIBIOBUF control statement ...................... 55
  DBRLBLUF control statement ...................... 56
  EXTENT control statement ......................... 56
  FRSPC control statement ............................ 57
  GDGBASE control statement ....................... 57
  GDGLIMIT control statement ...................... 57
  HPIO control statement ................................ 58
  ILDSBLD control statement ....................... 58
  ILEDATAC control statement ...................... 59
  ILESTORC control statement ...................... 60
  INPUT control statement ............................ 60
  ITKBLOAD control statement ...................... 61
  ITKBSSRVR control statement ...................... 61
  LPART control statement ............................. 62
  OADSPR control statement ............................ 63
  PAD control statement ................................ 64
  PARTINIT control statement ...................... 65
  PARTSTAT control statement ...................... 66
  PROGMON control statement ....................... 66
  RAFORMAT control statement ....................... 66
  RECNUM control statement ............................ 67
  REPAIRILK control statement ...................... 67
  RMSEGSZ control statement ............................ 68
  SEARCH control statement ............................ 68
  SEQERROR control statement ....................... 69
  SORT control statement .............................. 70
  SORTHIDAM control statement ..................... 72
  TWINSTAT control statement ....................... 72
  USEREXIT control statement ....................... 73
  USERHDR control statement ......................... 73
  USERHDRX control statement ....................... 75
  WFDNSN control statement ............................. 75
  WFIREC control statement ............................. 76
  WFISPAC control statement ............................. 76
  WFIUNIT control statement ............................. 77
  WFIUCosal control statement ....................... 77
  WFPHILQ control statement ............................. 78
  WFSCTRL control statement ............................. 78
  WFPUNIT control statement ............................. 79
  WFPVOL control statement ............................. 79
  WFPVOLCNT control statement ...................... 80
Reports and statistics .................................. 81
SYSPRINT data set ..................................... 81
SNAP data set ........................................ 103
HPSRAMSO data set ..................................... 103

Chapter 4. Reorganizing a HALDB database .................... 107
Scenario: Reorganizing a HALDB database ............... 108
Tips for reorganizing partitions individually .......... 110

Chapter 5. Reloading user-defined header format records ............. 111
Restrictions for reloading user-defined header format records .................. 112
Programming interface of user header conversion routine .................. 113
Chapter 6. Reloading a corrupted database ........................................ 117

Chapter 7. User exit interface ......................................................... 119
Parameters for the user exit interface ............................................. 121
Interface of Assembler user exit routine ......................................... 124
Interface of Enterprise COBOL for z/OS program user exit routine .......... 125
Interface of Enterprise PL/I for z/OS user exit routine ......................... 126

Chapter 8. Examples .................................................................. 127
Example 1: As a part of the reorganization process .......................... 128
Example 2: Creating an empty database ........................................... 130
Example 3: Reorganizing a HALDB database by using IMS HP Unload and IMS HP Load ............................... 131
Example 4: Creating WFP work data sets to rebuild PSINDEXes .............. 133

Part 3. Application programming interface for IMS High Performance Load ......................................................... 135

Chapter 9. Introduction to the Load API ............................................ 137

Chapter 10. Operating instructions for using the Load API ................. 139
Supported language and environment for Load API ............................. 140
Load API restrictions ....................................................................... 141
Load API data flow ......................................................................... 142
Load API JCL requirements ............................................................... 143
Load API control statements ............................................................. 146
  LAIPIPCB control statement ......................................................... 146
  LAIPIPCBNO control statement .................................................... 146
  LAIPIPDBD control statement ....................................................... 147
  PIPEOWAIT control statement .................................................... 147
  TRACE control statement ............................................................ 147
Load API output ............................................................................. 149
Status codes .................................................................................. 149
Trace Output report ........................................................................ 150
Application Abend Snap report ....................................................... 150

Part 4. Physical Sequence Sort for Reload ...................................... 153

Chapter 11. Introduction to Physical Sequence Sort for Reload .......... 155
Program functions of PSSR ............................................................... 156
Typical benefits of PSSR ................................................................. 159

Chapter 12. Sorting an unloaded database data set with PSSR .......... 161
Restrictions of PSSR ....................................................................... 162
Program structure and data flow of PSSR ......................................... 163
Reorganizing a database and using PSSR to sort the unloaded data set .......... 166
PSSR JCL requirements ................................................................... 170
PSSR FABSPSSR input .................................................................... 175
  FABSPSSR SYSIN data set .......................................................... 175
  FABSPSSR PSSRIN data set .......................................................... 177
  FABSPSSR CTL data set ............................................................... 178
PSSR output. .................................................................................. 190
  SORTOUT data set for non-HALDB processing .......................... 191
  SORTOUT2 data set for non-HALDB processing ....................... 191
  SORTOUT data set for HALDB processing ............................... 192
  SORTOUT2 data set for HALDB processing ............................. 192
  SORTOUT3 data set .................................................................. 193
  SORTOUT4 data set ................................................................. 193
  SORTOUTD data set ............................................................... 193
  Output partition data set .......................................................... 194
  SYSOUT data set .................................................................. 194
  SYSPRINT data set ................................................................. 194
  PR8 data set .................................................................. 194
  PR9 data set .................................................................. 198
  PR9X data set .................................................................. 199
  PR10 data set ................................................................ 200

Chapter 13. PSSR user exit ............................................................... 201
Program functions of PSSR user exit .............................................. 202
Parameters for PSSR user exit ....................................................... 204
Interface of Assembler user exit routine ......................................... 207
Interface of Enterprise COBOL for z/OS program user exit routine .......... 208
Interface of Enterprise PL/I for z/OS user exit routine ......................... 209

Chapter 14. PSSR usage examples .................................................. 211
Example 1: Normal reorganization (Changing root addressable area) ........ 212
Example 2: Preventing cascading .................................................... 215
Example 3: Iterative tuning analysis .................................................. 219
Example 4: Print reports about record sizes ...................................... 222
Example 5: HALDB processing ....................................................... 224

Part 5. Bitmap Resetter .................................................................. 225

Chapter 15. Introduction to the Bitmap Resetter utility ..................... 227

Chapter 16. Resetting bitmaps with the Bitmap Resetter utility .......... 229
Resetting bitmaps ....................................................................... 230
Bitmap Resetter JCL requirements ................................................ 231
Bitmap Resetter control statements .............................................. 234
  BMSETDSSG control statement .................................................. 234
  BMSETPART control statement .................................................. 235
Bitmap Resetter reports .................................................................. 236
  HPSIN Control Statements report .............................................. 236
  Parameter and DBD/Data Sets report ........................................... 236
About this information

IBM® IMS™ High Performance Load for z/OS® (also referred to as IMS High Performance Load or IMS HP Load) is a reloading tool that you can use to reload IMS full-function databases.

IMS High Performance Load is a component of the IBM IMS Database Solution Pack for z/OS. IMS High Performance Load is also available as a separately orderable product.

This information is designed to help IMS system programmers, application programmers, system analysts, database administrators, and computer operators who need to know how to operate IMS HP Load. It is a complete reference for this utility and covers the following topics:

• Understand the functions of IMS HP Load
• Run and use IMS HP Load after it is installed
• Use DD statements to control how you use IMS HP Load
• Run and use IMS HP Load reports
• Understand and recover from IMS HP Load problems

To use this book, you should have a working knowledge of:

• The z/OS operating system
• ISPF
• SMP/E

Also before using this guide, you should understand basic IMS concepts, the IMS environment, and your installation’s IMS system.

Always check the IMS Tools Product Documentation page for the most current version of this information:

Part 1. IMS High Performance Load overview

IBM IMS High Performance Load for z/OS (also referred to as IMS High Performance Load or IMS HP Load) provides a complete set of high performance reorganization reload procedures for various database organizations.

Topics:

- Chapter 1, “Introduction to IMS High Performance Load,” on page 3
- Chapter 2, “Installing and configuring IMS High Performance Load,” on page 23
Chapter 1. Introduction to IMS High Performance Load

IMS High Performance Load provides high-speed loading of IMS databases and a complete set of high-speed reorganization procedures. IMS High Performance Load saves you time and money by reducing the CPU and elapsed time that is required for IMS database loading and reorganization.

Topics:
- “What's new in IMS High Performance Load” on page 4
- “What does IMS High Performance Load do?” on page 9
- “Features and benefits” on page 12
- “Restrictions” on page 13
- “What's new in Version 2” on page 14
- “Integration with other utilities and products” on page 15
- “Compatibility with former products” on page 16
- “Utilities management solutions” on page 18
- “Service updates and support information” on page 19
- “IMS HP Load documentation and updates” on page 20
- “Accessibility features for IMS HP Load” on page 21
What's new in IMS High Performance Load

This topic summarizes the technical changes for this edition. Changes include publication updates and product enhancements.

New and changed information is indicated by a vertical bar (|) to the left of a change. Editorial changes that have no technical significance are not noted.

SC18-9222-07 (June 2016)

This edition covers the functional changes that are provided by the following APARs:

- PI47265, PI50574, PI51537, and PI53313

Major changes are as follows:

**APAR PI50574**

You can now sort PHIDAM database before loading the database. For more information, see “SORTIDAM control statement” on page 72.

**APARs PI51537 and PI53313**

You can use the new COMPAUTH control statement to call the segment compression exit in supervisor state. For more information, see “COMPAUTH control statement” on page 52.

SC18-9222-06 (February 2015)

This edition covers the functional changes that are provided by the following APARs:

- PM77018, PM79675, PM82937, PM87663, PM93095, PM94706, PM94789, PI05052, PI08722, PI14950, PI17977, PI18626, PI24722, PI27637, and PI30089

Major changes are as follows:

**APAR PM79675**

You can use the new TWINSTAT control statement to print statistics about the twin chained segments in the Segment Statistics report. For more information, see “TWINSTAT control statement” on page 72.

**APARs PM82937 and PI05052**

You can use the new PAD control statement to specify the padding character to fill out the segments to adjust to the length defined in the DBD definition. For more information, see “PAD control statement” on page 64.

**APAR PM94789**

- This APAR enhances IMS HP Load to enable the Smart Reorg utility of IMS Database Reorganization Expert rebuild partitioned secondary indexes (PSINDEXes).
- You can use the new INITONLY parameter for the ILDSBLD control statement. If you specify this parameter, IMS HP Load initializes ILDSs without building ILEs. For more information, see “ILDSBLD control statement” on page 58.

**APAR PI14950**

You can use the new CHKEMPTYUL control statement to check whether the unloaded data set contains segment data. You can also define the
return code that is set when no segment data is found in the unloaded data set. For more information, see “CHKEMPTYUL control statement” on page 50.

APAR PI27637
Support for IMS Version 14 Release 1 has been added.

SC18-9222-05

This edition covers the functional changes that are provided by the following APARs: PK97074, PK96383, PM00732, PM01857, PM07578, PM21339, PM22118, PM28920, PM31499, PM43460, PM39207, PM46445, PM47889, PM52305, PM57126, PM58706, PM61441, PM60645, PM61213, PM62216, PM65263, and PM68882.

Major changes are as follows:

**APARs PM00732 and PM58706**
IMS HP Load has been enhanced to support Tools Base IMS Tools Knowledge Base. You can store the reports that are generated by IMS HP Load in the Output repository of IMS Tools Knowledge Base. For more information, see “Configuring the IMS Tools Knowledge Base server” on page 25.

**APAR PM07578**
You can create a generation data group (GDG) base entry when DFSURWF1 is defined as a generation data set and when a GDG base does not exist. For more information, see “GDGBASE control statement” on page 57.

**APARs PM22118 and PM65263**
Support for IMS Version 12 Release 1 has been added.

**APAR PM46445**
DDEFPDS DD statement has been supported. You can use this DD statement to specify the IDCAMS commands to delete or to define database data sets while you reload the database. For more information, see “JCL requirements” on page 35.

**APAR PM62216**
You can create WFP data sets, which are work data sets for rebuilding PSINDEXes with IMS Index Builder. By using the WFP data sets as input to IMS Index Builder, you can rebuild PSINDEXes without scanning the database. For more information, see HPSRFPnnn DD statement in “JCL requirements” on page 35.

**APAR PM68882**
You can use the REPAIRILK control statement to reassign ILKs. By reassigning ILKs, you can prevent duplicate segment ILKs. For more information, see “REPAIRILK control statement” on page 67.

SC18-9222-04

This edition covers the functional changes that are provided by the following APARs: PK64733, PK66948, PK71096, PK72387, PK73015, PK74299, PK75509, PK76659, PK78922, PK88818, and PK95918.

Major changes are as follows:

**APAR PK73015**
Two new control statements are added:
• The BMSETDSG control statement specifies the threshold value for the bitmap block. For more information, see “BMSETDSG control statement” on page 49.

• The WF1REC control statement specifies whether to write the WF1 records in the DFSURWF1 or HPSRSINDEX output work data set. This control statement is useful when you do not need the WF1 records for the secondary index or logical relationships. For more information, see “WF1REC control statement” on page 76.

APAR PK74299
Support for IMS Version 11 Release 1 has been added.

APAR PK75509
The SORTHIDAM control statement is added. This control statement specifies whether to run the sort processing before the load process starts for an HIDAM database. For more information, see “SORTHIDAM control statement” on page 72.

APAR PK88818
The following enhancements are made:
• The CHKTRAIL control statement is added. This control statement specifies whether to check if a trailer record exists in the "HD or "SH format of the unload data set. For more information, see “CHKTRAIL control statement” on page 51.
• Support for installation default options is added. For more information, see Chapter 17, “Setting installation default options,” on page 241.

APAR PK95918
*CP format (the communication industry partitioned format) is supported as an input for reload.

SC18-9222-03

This edition covers the functional changes that are provided by the following APARs: PK37945, PK42361, PK42845, PK44725, PK46331, PK46593, PK48420, PK52055, PK54068, PK57506, PK58296, and PK59570.

Major changes are as follows:
APAR PK48420
A new option is added to the PARTINIT control statement. By enabling this option, you do not need to set the PARTITION INIT NEEDED flag in the DBRC RECON for the partitions that do not contain any segment after reload.

APAR PK54068
This APAR enables the IPR Reload utility to store several IPR Reload utility reports in the output repository of IMS Tools Knowledge Base.

APAR PK57506
A new option is added to the ILDSBLD control statement. This option enables the rebuild of ILDS by using less data space than ILDSBLD=YES.

APAR PK59570
This APAR enables the validation of changes in the compression routine during reorganization.
SC18-9222-02

This edition covers the functional changes that are provided by the following APARs: PK16004, PK18989, PK21494, PK28098, PK33129, and PK34472.

Major changes are as follows:

**APARs PK16004, PK18989**  
The OADSPR control statement is provided to reduce the usage of data space when the overflow area of (P)HDAM is processed.

**APAR PK21494**  
A new option COND, is added to the SORT control statement. With this option, you can run the physical sequential sorting depending on conditions.

**APAR PK28098**  
Support for IMS Parallel Reorganization Version 3 Release 2 has been added.

**APAR PK33129**  
Support for IMS Version 10 Release 1 has been added.

**APAR PK34472**  
The SORT control statement is supported for the reload task of the IPR Driver. To use this function, APAR PK33846 must be applied to IMS Parallel Reorganization Version 3 Release 2.

SC18-9222-01

This edition covers the functional changes that are provided by the following APARs: PQ97799, PK07435, PK09141, PK10654, PK10446, PK11532, and PK12257.

Major enhancements are as follows:

**APARs PK07435, PK09141**  
These APARs provide the LOAD API interface. It makes the initial load process much faster than the DL/I load the database and provides the statistics about the database. These APARs also provide a new control statement SORT to specify whether to run the physical sequential sort processing before the load process starts.

**APAR PK10446**  
This APAR provides the Bitmap Resetter function. Bitmap Resetter is a utility that changes the bitmap data.

**APAR PK12257**  
This APAR supports a new control statement, CHKUPDT, to enable IMS HP Load do a check against DBRC to see whether an update of the database has been done between unload and reload processes.

**Changes in Version 1**

The following items describe the publication updates and product enhancements of IMS High Performance Load for OS/390®, Version 1.

SC27-0938-01

This edition incorporates documentation changes that are made in the following APARs for IBM IMS High Performance Load for OS/390:
Overall:

- APAR PQ57582: Support to run under IMS Version 8

For the Load utility:

- APAR PQ47222: Restrictions added for the utility
- APAR PQ51000: Description of DATASPACE=YES changed and its message added
- APAR PQ51495: Message HPSR3695E added
- APAR PQ53316: Support for HISAM and SHISAM
- APAR PQ54113: Support for HISAM and SHISAM
- APAR PQ54734: Description added for the user exit interface
- APAR PQ59763: Description added for COMP=
- APAR PQ59841: Support for the DECODE option of IMS Parallel Reorganization for z/OS Version 2
- APAR PQ61147: Message HPSR3674E added
- APAR PQ62856: Support for SEQERROR=ACCEPT
- APAR PQ69009: Support for user header conversion exit

For PSSR:

- APAR PQ48740: Support for HIDAM
- APAR PQ52914: Description for key compress segments added
- APAR PQ61148: Message FAS3642E added
- APAR PQ67346: Description of the SORT parameter corrected
- APAR PQ67957: Support for the PAD parameter of the Segment Edit/Compression exit routine

SC27-0938-00

This edition covers IMS High Performance Load for OS/390 Release 1. The product is a follow-on of the IMS System Utilities/Data Base Tools (DBT) Version 2 Fast Reorganization Reload and HDAM Physical Sequence Sort/Reload. This User's Guide contains new information about the following subjects:

- Support for IMS Version 7, including High Availability Large Database (HALDB)
- The following new functions are introduced for Physical Sequence Sort for Reload:
  - User Exit support
  - Segment Edit/Compression exit support
  - IMS/ESA® Year 2000 Exit Tool (5697-E04) support
What does IMS High Performance Load do?

IMS High Performance Load consists of the IMS HP Load utility, the Physical Sequence Sort for Reload utility, and the Bitmap Resetter utility. It also provides the Load application programming interface (Load API).

IMS HP Load provides a complete set of high performance reorganization reload procedures for the following database organizations:

- HDAM (Hierarchical Direct Access Method)
- HIDAM (Hierarchical Indexed Direct Access Method)
- HISAM (Hierarchical Indexed Sequential Access Method)
- SHISAM (Simple Hierarchical Indexed Sequential Access Method)
- PHDAM (Partitioned Hierarchical Direct Access Method)
- PHIDAM (Partitioned Hierarchical Indexed Direct Access Method)

**IMS HP Load utility**

The IMS HP Load utility provides high performance reload processing for databases. The utility is a performance replacement of the IMS HD Reorganization Reload utility (DFSURGL0). The IMS HP Load utility supports various database organizations. It creates the indirect list data set (ILDS) of a HALDB that is compatible with the one created by IMS standard IMS HD Reorganization Reload utility. Also, the utility provides reports that contain statistics about space use and segment pointers.

**Load API (application programming interface)**

IMS High Performance Load provides the Load application programming interface (Load API). By using the Load API, you can run the initial load process much faster than that of IMS DL/I.

**Physical Sequence Sort for Reload utility**

The Physical Sequence Sort for Reload utility sorts an unloaded database data set that is produced for an HDAM database to reduce the elapsed time that is required to reload the database. It also sorts an unloaded database data set for an HDAM, a PHDAM, a HIDAM, a PHIDAM, a HISAM, or a SHISAM database. When you change the partition selection method of a HALDB, Physical Sequence Sort for Reload is required before you reload the HALDB by using IMS HP Load. The utility also provides reports, for use in the tuning analysis of the randomizing parameters.

**Bitmap Resetter utility**

The Bitmap Resetter utility can be used to adjust the bitmap of an HDAM, HIDAM, PHDAM, or PHIDAM database to accommodate denser packing of the database blocks.

The following subsections describe the program functions of the IMS HP Load utility.

- “Reloading IMS databases by using high performance methods” on page 10
- “Initializing an empty database” on page 10
- “Providing a user exit facility for additional processing of each segment” on page 10
- “Generating statistics reports to aid in tuning the database” on page 11
Reloading IMS databases by using high performance methods

IMS HP Load performs high-performance reloading of IMS databases from an HD unloaded data set that is created by:

- Unload utilities of IMS High Performance Unload for z/OS
- IPR Unload utility of IMS Database Reorganization Expert for z/OS (formerly known as IMS Parallel Reorganization for z/OS)
- IMS HD ReorganizationUnload utility (DFSURGU0)

You can use this utility to increase the efficiency of the database reorganization process.

When you use IMS HP Load to reload a HALDB, the partition data set of that HALDB does not need to be initialized. You can skip the step before you run IMS HP Load, by specifying the PARTINIT=YES statement in the FRRIN data set. IMS HP Load itself performs the required partition initialization.

Optionally, IMS HP Load can reload an HD unloaded database data set with compressed segments, created by the IMS High Performance Unload for z/OS unload utilities or by other compatible utilities, in a compressed form without calling the segment compression routine. In this case, the segment compression routine must not be changed during the reorganization. IBM IMS Library Integrity Utilities for z/OS can be used to detect such a change. If it detects the change and the input unloaded data set is in a compressed form, the reload job ends abnormally. For information about IMS Library Integrity Utilities, see the IMS Library Integrity Utilities User’s Guide.

IMS HP Load supports logical relationships and secondary indexes. Logical relationships and secondary indexes of HALDB databases are also supported.

IMS HP Load supports the IMS DBRC feature.

Initializing an empty database

If you specify the DFSUINPT DD DUMMY statement, HDAM and HIDAM databases are initialized.

If you specify the DFSUINPT DD DUMMY statement and the PARTINIT=YES control statement, and also if the partition initialization flag is on for the partition to initialize, PHDAM and PHIDAM databases are initialized.

For an HDAM database and each partition of a PHDAM database, the entire root addressable area is preformatted.

An HIDAM and each partition in a PHIDAM database contains a root segment with a key of all X’FF’.

Providing a user exit facility for additional processing of each segment

IMS HP Load supports a user exit routine. IMS HP Load invokes the user exit routine to exercise additional selectivity or modification of segments to be received from the HD unloaded data set.
Generating statistics reports to aid in tuning the database

IMS HP Load generates statistics reports about data sets, segments, and segment pointers. The reports serve as a valuable aid in tuning the database.
Features and benefits

IMS High Performance Load offers several unique and significant features that you can use to increase the efficiency of reloading your databases.

High performance
IMS HP Load is designed to provide high-performance processing by default parameter settings. You can get optimized performance on each reorganization job.

Ease of use
By dynamically allocating a database data set for a full-function database and HALDB, IMS HP Load saves you the trouble of writing DD statements for that database data set.

Data compatibility
Data that are created by IMS HP Load, such as the DFSURWF data set, are compatible with IMS standard utilities.

Saving time
IMS HP Load does not require HALDB partition data set initialization before reloading. For partitions not populated by any segment during reload, IMS HP Load initializes the partition data set.

IMS HP Load offers the following benefits:
• Reduces time and costs for reorganization IMS databases
• Offers more flexible control over the reorganization of IMS databases
• Provides a means to initialize empty HDAM, HIDAM, HISAM, or SHISAM databases
• Initializes partitions of PHDAM or PHIDAM databases within its reload processing.
• Provides statistical data that is needed for database tuning tasks and space analysis tasks.
Restrictions

The following restrictions apply to all the utilities of IMS HP Load.

- IMS HP Load does not run under IMS Utility Control Facility (UCF).
- IMS HP Load does not support the checkpoint and restart capability.
- No user-written data conversion exit routine is supported (see “Software requirements” on page 24).

Other restrictions for each utility are described in each utility topic.
What's new in Version 2

IMS High Performance Load Version 2 has been enhanced with features that simplify the reloading of IMS databases. This topic describes the enhancements included in IMS High Performance Load for z/OS Version 2. It also summarizes the major changes that are made by the APARs for IMS High Performance Load for OS/390 Version 1.

Provides the HPIO control statement for using High Performance I/O Driver.
You can specify whether to use High Performance I/O Driver for output database data sets. For more information, see "HPIO control statement" on page 58.

Provides the USERHDR control statement for unloaded files that have customized segment header.
You can reload the unloaded data set that has customized segment header by using the USERHDR control statement. For more information, see "Control statements" on page 45.

Provides ease of use by the dynamically allocating the database data set.
If HPIO=YES is specified, the authorization of Authorized Program Facility (APF) is required for the load library of IMS HP Load. For specifying libraries that contain the members for dynamic allocation, which are generated by using the DFSMDA macro of IMS, a new DD statement IMSDALIB is supported. It bypasses the step that needs APF-authorization of the dynamic allocation library.

For more information about the specification of the IMSDALIB DD statement, see "JCL requirements" on page 35.

Eliminates the POINTERS DD statement.
This DD statement is not necessary in Version 2. IMS HP Load uses data space as a temporary space for database blocks and pointer information instead of using the DD statements. If any is coded in a JCL stream for Version 2, it is ignored. For more information, see "Compatibility with former products" on page 16.

Provides the SORT control statement.
The HDAM/PHDAM unloaded data must be in the physical sequence of the block and RAP numbers. With this control statement, you can reload without PSSR even if the HDAM or the PHDAM database is not in physical sequence.

If you reload a database by using IMS High Performance Load for z/OS Version 2 Release 1, you cannot use PSSR to prevent cascading. To prevent cascading and avoid degradation of performance in reloading, it is recommended that you use the SORT=YES and the SEARCH=-nnn parameter instead of using PSSR.

For example, if the randomizer is changed, you can use this control statement by specifying SORT=YES.

Provides the Load API.
You can use the application program as input of the Load API for the initial load processing. For more information, see Chapter 9, "Introduction to the Load API," on page 137.
Integration with other utilities and products

IMS HP Load interfaces with other IMS utilities and products.

IMS HP Load receives the HD unloaded data set that is created by the following utilities:
- IMS High Performance Unload for z/OS unload utilities
- IPR Unload utility of IMS Database Reorganization Expert for z/OS
- IMS HD Reorganization Unload utility (DFSURGU0)

IMS HP Load also receives, compressed format, the HD unloaded data set created by the following utilities:
- IMS High Performance Unload for z/OS unload utilities
- IPR Unload utility of IMS Database Reorganization Expert for z/OS
- IMS DBT Version 2 High Speed Sequential Retrieval (HSSR) unload utilities

**Note:** If you create an unloaded data set by specifying the DECN option without applying APAR PQ22654, you must specify the COMPRESS=NO parameter when you reload the data set.

The *work file 1 data set* (DFSURWF1) is compatible with the one that is created by the IMS HD Reorganization Reload utility.

IMS HP Load integrates with IMS Tools Knowledge Base for storing and retrieving reports. For more information, see "Configuring the IMS Tools Knowledge Base server" on page 25.
Compatibility with former products

IMS High Performance Load for z/OS Version 2 Release 1 is compatible with its former products. The following topics explain the compatibility of IMS High Performance Load for z/OS, Version 2 Release 1 with its earlier products.

Compatibility with IMS High Performance Load for OS/390 Version 1

IMS High Performance Load for z/OS Version 2 Release 1 is compatible with IMS High Performance Load for OS/390 Version 1.

The following sections explain compatibility with IMS High Performance Load for OS/390, Version 1.

Subsections:
- “IMS High Performance Load”
- “Physical Sequence Sort for Reload (PSSR)” on page 17

IMS High Performance Load

JCL and control statements that are used in Version 1 can be used in Version 2, except that:

The SORT control statement must be added if you change the randomizer for HDAM or PHDAM.

If the randomizer or its parameter is changed between the unload process and the reload process, either the control statement SORT=YES or SORT=COND is needed because of a new restriction that HDAM or PHDAM unloaded data must be in the physical sequence of the block and RAP numbers. This change eliminates direct I/O and improves performance.

Without the control statement SORT=YES or SORT=COND, the error message HPSR3630E RAP ASSIGNED OUT OF PHYSICAL SEQUENCE is issued. For more information about the SORT control statement, see “SORT control statement” on page 70.

The DATASPACE control statement is not effective and the data space is always used. The DATASPACE control statement is not effective. If specified, it is ignored. IMS High Performance Load Version 2 Release 1 or later uses data space by its default, instead of using data sets POINTERS DD or OVERFLOW DD, to keep the pointer information and the block image of the overflow area.

If there is not enough data space available on the system, and the error HPSR3643E DSPSERV CREATE or EXTEND FAILURE OCCURRED is shown, you can reduce the usage of the data space by specifying OADSPR=YES. For more information about the OADSPR control statement, see “OADSPR control statement” on page 63.

If you specify ILDSBLD=YES, you can also reduce the usage of data space by specifying ILDSBLD=YES,DSPR. For more information about the ILDSBLD control statement, see “ILDSBLD control statement” on page 58.

The condition for how to use the OVERFLOW DD statement is changed.

The OVERFLOW DD is used only if OADSPR=YES and HPIO=NO is specified. Otherwise, data space is used instead of using the data set.
The POINTERS DD statement is not effective.
The POINTERS DD is no longer used. If this DD is specified, it is ignored.

**Physical Sequence Sort for Reload (PSSR)**

Before you use PSSR, the following differences must be noted:

- If you reload an HDAM by IMS HP Load Version 2 Release 1 or later, the SORTOUT2 DD cannot be used. For the details, see “Considerations before using PSSR” on page 155.
- You cannot use PSSR to prevent cascading. To prevent cascading and avoid degradation of performance in reloading, you can use the SORT=YES and the SEARCH=-nnn parameter instead of using PSSR.

**Compatibility with IMS DBT Version 2**

IMS High Performance Load for z/OS is compatible with IMS System Utilities/Data Base Tools (IMS DBT) Version 2 Fast Reorganization Reload and HDAM Physical Sequence Sort/Reload.

The load module name DFSURGL0 of IMS DBT Version 2 Fast Reorganization Reload can be used in IMS HP Load. The JCL statement for IMS DBT Version 2 Fast Reorganization Reload can be used for IMS HP Load.
Utilities management solutions

IBM solutions help IT organizations maximize their investment in IMS databases while staying on top of some of today’s toughest IT challenges. Utilities management solutions can help you achieve higher availability and better performance during database maintenance while enhancing the productivity of both database administrators and system programmers.

Underlying the operation of any database management system are the utilities. With the number of database objects that are growing exponentially, especially when dealing with ERP applications such as SAP, the impact of managing utility jobs, meeting service level agreements, and ensuring recoverability can be overwhelming.

IBM offers IMS Tools that help the Utilities Management process for example:

In an on-demand world 24x7 data availability is a key requirement. Reorganization tools from IBM can help with the performance of key functions such as unloading and reloading IMS data, without affecting data access.

These and other IMS Tools can help you achieve higher availability and better performance during data maintenance while enhancing the productivity of both database administrators and system programmers.

Many IMS Tools products provide database management features that are not available in IMS itself or that provide enhancements to capabilities that are built into IMS.

IBM IMS High Performance Load for z/OS is only one of several IMS Tools products that provide enhancements to the process of utility management operations for your databases.

For example, IMS HP Load provides a high performance database reloading capability for IMS full-function and HALDB databases. With the latest version of IMS HP Load, Version 2.1, database reloading performance is improved to reduce CPU time utilization, improving I/O processing to reduce I/O wait and using the High Performance I/O driver. IMS HP Load now supports IMS Database Reorganization Expert for z/OS capabilities such as image copy creation during database reorganization. IMS HP Load works with IMS High Performance Unload to help you to load compressed data that is previously unloaded in a compressed format.

Other IMS Tools products that can assist with database utilities management include:
- IMS Cloning Tool
- IMS Database Control Suite
- IMS Database Reorganization Expert
- IMS Database Solution Pack
- IMS Fast Path Solution Pack
Service updates and support information

Service updates and support information for this product, including software fix packs, PTFs, frequently asked questions (FAQs), technical notes, troubleshooting information, and downloads, are available from the web.

To find service updates and support information, see the following website:

IMS HP Load documentation and updates

IMS Tools information is available at multiple places on the web. You can receive updates to IMS Tools information automatically by registering with the IBM My Notifications service.

Information on the web

The IMS Tools Product Documentation web page provides current product documentation that you can view, print, and download. To locate publications with the most up-to-date information, refer to the following web page:


IBM Redbooks® publications that cover IMS Tools are available from the following web page:

http://www.redbooks.ibm.com

The Data Management Tools Solutions website shows how IBM solutions can help IT organizations maximize their investment in IMS databases while staying ahead of today’s top data management challenges:


Receiving documentation updates automatically

To automatically receive emails that notify you when new technote documents are released, when existing product documentation is updated, and when new product documentation is available, you can register with the IBM My Notifications service. You can customize the service so that you receive information about only those IBM products that you specify.

To register with the My Notifications service:

1. Go to http://www.ibm.com/support/mysupport
2. Enter your IBM ID and password, or create one by clicking register now.
3. When the My Notifications page is displayed, click Subscribe to select those products that you want to receive information updates about. The IMS Tools option is located under Software > Information Management.
4. Click Continue to specify the types of updates that you want to receive.
5. Click Submit to save your profile.

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 book or any other IBM product documentation, use one of the following options:

• Use the online reader comment form, which is located at http://www.ibm.com/software/data/rcf/
• Send your comments by email to comments@us.ibm.com. Include the name of the book, the part number of the book, the version of the product that you are using, and, if applicable, the specific location of the text you are commenting on, for example, a page number or table number.
Accessibility features for IMS HP Load

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use information technology products successfully.

Accessibility features

The major accessibility feature of the product is the keyboard-only operation for ISPF editors. It uses the standard TSO/ISPF interface.

Keyboard navigation

You can access IMS ISPF panel functions by using keyboard shortcuts.

For information about navigating the IMS ISPF panels by using TSO/E or ISPF, refer to the following publications for information about accessing ISPF interfaces:

- z/OS ISPF User’s Guide, Volume 1, SC34-4822
- z/OS TSO/E Primer, SA22-7787
- z/OS TSO/E User’s Guide, SA22-7794

These publications describe how to use ISPF, including the use of keyboard shortcuts or function keys (PF keys), include the default settings for the PF keys, and explain how to modify their functions.
Chapter 2. Installing and configuring IMS High Performance Load

The following topics provide the prerequisites and requirements for installing IMS High Performance Load, and explain the configuration procedures.

Topics:

- “Hardware and software prerequisites” on page 24
- “Configuring the IMS Tools Knowledge Base server” on page 25
- “Configuring for DBRC command authorization” on page 27
- “Configuring the environment to use the Load API” on page 28
Hardware and software prerequisites

Before you install and configure IMS HP Load, make sure that your environment meets the following minimum hardware and software requirements.

Software requirements

IMS High Performance Load for z/OS operates in a z/OS environment. Its operating system requirements are the same as those for IMS supported by IMS High Performance Load for z/OS.

IMS High Performance Load for z/OS requires one of the currently supported versions of IMS or IMS Database Value Unit Edition.

Note: In these topics, all supported versions of IMS are referred to as IMS, except where distinctions among them need to be made.

IMS High Performance Load for z/OS supports the data conversion exit of IMS. For this support, it requires IMS/ESA Year 2000 Exit Tool Release 1 (5697-E04).

IMS High Performance Load for z/OS supports only the exit routines for data conversion that are provided by this product.

To store IMS HP Load reports in the IMS Tools Knowledge Base server (ITKBSRVR=server_name option), IBM Tools Base IMS Tools Knowledge Base for z/OS, Version 1 Release 4 or later must be installed.

Physical Sequence Sort for Reload requires DFSORT (Data Facility Sort), which is a part of z/OS. It is the same as when you specify YES for the SORT control statement.

To write the user exit routine, use:
- Enterprise COBOL for z/OS
- Enterprise PL/I for z/OS

To use the user exit routine that is written in one of these languages, you must use it in Language Environment®.

Hardware requirements

The hardware requirements are the same as those for IMS.
Configuring the IMS Tools Knowledge Base server

IBM Tools Base IMS Tools Knowledge Base (IMS Tools KB) is a Tools Base component that provides common services for storing and viewing reports that are generated by other participating IMS Tools products. Configure the IMS Tools Knowledge Base server if you want to store IMS HP Load reports in the IMS Tools Knowledge Base Output repository.

About this task

To fully participate in the IMS Tools Knowledge Base information management environment, each IMS tool must be enabled to communicate with the IMS Tools Knowledge Base server. An enabled IMS tool can automatically send its generated reports to the IMS Tools Knowledge Base repository. This version of IMS HP Load is enabled to participate in the IMS Tools Knowledge Base environment.

Procedure

To store IMS HP Load reports in the Output repository of IMS Tools KB by using the ITKBSRVR=server_name control statement, you must set up the environment by completing the following steps:

**Note:** For detailed procedures about the tasks that are related to IMS Tools Knowledge Base, see the following publications:

- For Tools Base 1.5 or later, *Tools Base Configuration Guide*

1. Ensure that APAR PM00732 is applied to IMS HP Load.
2. Ensure that Tools Base IMS Tools Knowledge Base is installed.
3. Set up an IMS Tools KB server.
4. If needed, register the RECON.
5. Register IMS HP Load with IMS Tools Knowledge Base.

   The registration process defines IMS HP Load reports to IMS Tools Knowledge Base so that IMS HP Load reports can be stored in the IMS Tools Knowledge Base repository.

   Use the IMS Tools Knowledge Base product administration utility (HKTAPRA0) to register IMS HP Load with IMS Tools Knowledge Base. The product definition table for registering IMS HP Load is HPSRITKB. For more information, see the topic "Registering IMS Tools products by using the definition table of the product" in the following publications:

   - For Tools Base 1.5 or later, *Tools Base Configuration Guide*

   After you register IMS HP Load with IMS Tools Knowledge Base, see the topic "Listing registered products and reports" in the *Tools Base IMS Tools Knowledge Base User’s Guide* or the *Tools Base Configuration Guide* to list the registered products and reports, and ensure that IMS HP Load (product ID is DL) is added.

6. Verify communication with the IMS Tools Knowledge Base server.

   Verify that the systems that you will run IMS HP Load on will have XCF communications with the IMS Tools Knowledge Base server and that the FPQ subsystem is started on each of these systems. In the *Tools Base IMS Tools*
Knowledge Base User’s Guide or the Tools Base Configuration Guide, see the topic "Defining IMS Tools Knowledge Base to the operating system".

7. Request that IMS HP Load write reports to the IMS Tools Knowledge Base repository.

   Specify the IMS Tools Knowledge Base server information on the ITKBLOAD and ITKBSRVR control statements. These control statements can be specified in the FRRIN data set. These control statements direct IMS HP Load to communicate to the IMS Tools Knowledge Base server so that IMS HP Load reports can be stored in the IMS Tools Knowledge Base repository. For more information about the control statements to specify in the FRRIN data set and the reports that can be stored in the repository, see "ITKBSRVR control statement" on page 61 and "ITKBLOAD control statement" on page 61.

What to do next

After you complete these steps, IMS HP Load can store its reports in the Output repository of IMS Tools Knowledge Base. Before you run IMS HP Load jobs, always confirm that the IMS Tools Knowledge Base server is initialized.

When the reports are stored in the repository, you can use the ISPF user interface to view, print, and manage reports.
Configuring for DBRC command authorization

If IMS DBRC command and API request authorization support with RACF® is used, the user ID of the master address space must have authorization to access resources so that IMS HP Load can issue DBRC commands and DBRC API requests.

Procedure

The user ID under which IMS HP Load runs must be authorized to access resources for certain DBRC commands and API requests. Such resources are listed in the following table.

Table 1. DBRC command authorization for IMS High Performance Load

<table>
<thead>
<tr>
<th>Resource</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>hlq.LIST.DB.dbname</td>
<td>Is always issued.</td>
</tr>
<tr>
<td>hlq.LIST.RECON</td>
<td>Is issued only when processing a HALDB OSAM database.</td>
</tr>
<tr>
<td>hlq.STDBRC</td>
<td>Is issued only when processing a HALDB OSAM database.</td>
</tr>
</tbody>
</table>

For more information about authorizing user IDs, see the topic "Security for DBRC commands and API requests" in IMS System Administration.
Configuring the environment to use the Load API

If you want to use the application programming interface of IMS HP Load (Load API), configure the environment before you use the Load API.

Procedure

- Apply the PTF for the following APAR to your IMS environment:
  - For IMS Version 7, APAR PK03282.
  - For IMS Version 8, APAR PK03616.
  - For IMS Version 9, APAR PK03617.

Note: APARs are not required for IMS Version 10 or higher.

- Before you use the Load API for the first time, copy the module HPSRSVI0 in the member of IMS HP Load library to a library that is concatenated to the STEPLIB DD with the name DFSISVI0.

Tip: If you already created DFSISVI0 while preparing IMS High Performance Unload, you do not need to copy the HPSRSVI0 module again.
Part 2. IMS High Performance Load

The IMS HP Load utility replaces the IMS HD Reorganization Reload utility (DFSURGL0) to reload IMS full-function databases by using high-performance methods. The utility also analyzes database performance and prepares reports that contain statistics about space use and segment pointers.

Topics:
- Chapter 3, “Reloading IMS databases with IMS High Performance Load,” on page 31
- Chapter 4, “Reorganizing a HALDB database,” on page 107
- Chapter 5, “Reloading user-defined header format records,” on page 111
- Chapter 6, “Reloading a corrupted database,” on page 117
- Chapter 7, “User exit interface,” on page 119
- Chapter 8, “Examples,” on page 127
Chapter 3. Reloading IMS databases with IMS High Performance Load

The IMS HP Load utility is a performance replacement of the basic IMS HD Reorganization Reload utility (DFSURGL0). It reloads IMS HDAM, HIDAM, HISAM, SHISAM, PHDAM, and PHIDAM databases by using high-performance methods. It also analyzes database performance and prepares reports that contain statistics about space use and segment pointers.

Topics:

- “Data flow” on page 32
- “Restrictions” on page 33
- “Reloading a database” on page 34
- “JCL requirements” on page 35
- “Control statements” on page 45
- “Reports and statistics” on page 81
Data flow

The IMS HP Load utility reads data from the unloaded data set that is generated by the IMS HP Unload utility, the IPR Unload utility of IMS Database Reorganization Expert, or the IMS HD Reorganization Unload utility (DFSURGU0), and reloads the data into the specified database.

The behavior of the IMS HP Load job can be controlled by the control statements that are specified in the FRRIN data set. After the job is run, the data is reloaded into the specified database and SYSPRINT data set contains statistics reports about the database.

The following figure shows the general data flow for the IMS HP Load utility.

Figure 1. Data flow of IMS HP Load
Restrictions

IMS HP Load performs the same functions as the IMS HD Reorganization Reload utility, with certain restrictions.

The following restrictions apply to the IMS HP Load utility:

• HDAM DBDs must specify a nonzero maximum relative block number in the RMNAME operand of the DBD macro because it formats the entire root addressable area.

• IMS HP Load does not run under the Utility Control Facility (UCF). Checkpoint and restart of the reload process under the UCF are not supported.

• Because of the difference between the free space search algorithms of the IMS HD Reorganization Reload utility and of IMS HP Load, the latter might use more DASD.

• If you want to change partition boundaries during the reorganization of a HALDB, you must sort the unloaded records by specifying the SORT=YES statement in the FRRIN data set.

• If the access method used is VSAM, the total record number in the catalog information is cleared. But if you did not receive any warning or error message from IMS HP Load, all records are reloaded into the target database data sets.

• If the randomizer or its parameter is changed between unload and reload processes, you must sort the unload records by specifying the SORT=YES statement in the FRRIN data set.
Reloading a database

To reload data into a database, you must prepare JCL for the IMS HP Load job.

Procedure

1. Prepare JCL for IMS HP Load.
   For the JCL requirements, see "JCL requirements" on page 35.

2. Code the FRRIN control statements.
   FRRIN control statements control the behavior of the job. See "JCL requirements" on page 35 for descriptions of the FRRIN control statements.

3. Submit the JCL to run the job.

4. Check the output reports and messages.
   For more information about the reports and messages, see "Reports and statistics" on page 81.
JCL requirements

JCL required to run IMS HP Load is basically the same as for the IMS HD Reorganization utility, with the addition of a STEPLIB DD statement to some steps.

Subsections:
- “EXEC and DD statements”
- “Default values for the data set access method buffer” on page 44

EXEC and DD statements

The following table summarizes the DD statements. Explanations of JCL parameters follow the table.

Table 2. DD statements and record format for IMS HP Load

<table>
<thead>
<tr>
<th>DDNAME</th>
<th>Use</th>
<th>Format</th>
<th>Need</th>
</tr>
</thead>
<tbody>
<tr>
<td>STEPLIB</td>
<td>Input</td>
<td></td>
<td>Required</td>
</tr>
<tr>
<td>IMSDALIB</td>
<td>Input</td>
<td>Same as that used in the IMS</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>control region startup JCL</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>stream</td>
<td></td>
</tr>
<tr>
<td>IMS</td>
<td>Input</td>
<td></td>
<td>Required</td>
</tr>
<tr>
<td>PROCLIB</td>
<td>Input</td>
<td></td>
<td>Optional</td>
</tr>
<tr>
<td>DFSRESLB</td>
<td>Input</td>
<td></td>
<td>Optional</td>
</tr>
<tr>
<td>DFSUINPT</td>
<td>Input</td>
<td></td>
<td>Required</td>
</tr>
<tr>
<td>DFSURCDS</td>
<td>Input</td>
<td>LRECL=1600</td>
<td>Optional</td>
</tr>
<tr>
<td>DFSVSAMP</td>
<td>Input</td>
<td>LRECL=80</td>
<td>Optional</td>
</tr>
<tr>
<td>FRRIN</td>
<td>Input</td>
<td>LRECL=80</td>
<td>Optional</td>
</tr>
<tr>
<td>DDEFPDS</td>
<td>Input</td>
<td>RECFM=FB, LRECL=80</td>
<td>Optional</td>
</tr>
<tr>
<td>RECONx</td>
<td>Input/Output</td>
<td></td>
<td>Optional for Non-HALDB</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Required for HALDB</td>
</tr>
<tr>
<td>database</td>
<td>Output</td>
<td></td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>See Note 4</td>
</tr>
<tr>
<td>DFSURWF1</td>
<td>Output</td>
<td>RECFM=VB, LRECL=900</td>
<td>Optional</td>
</tr>
<tr>
<td>HPSRPnnn</td>
<td>Output</td>
<td>RECFM=FB, LRECL=7250</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>See Note 4</td>
</tr>
<tr>
<td>HPSRSIDX</td>
<td>Output</td>
<td>RECFM=VB, LRECL=900</td>
<td>Optional</td>
</tr>
<tr>
<td>OVERFLOW</td>
<td>Input/Output</td>
<td>RECFM=VB, LRECL=133</td>
<td>Optional</td>
</tr>
<tr>
<td>HPSRISMG</td>
<td>Output</td>
<td>LRECL=133</td>
<td>Optional</td>
</tr>
<tr>
<td>SEQERROR</td>
<td>Output</td>
<td>RECFM=VB</td>
<td>Optional</td>
</tr>
<tr>
<td>SYSPRINT</td>
<td>Output</td>
<td>LRECL=81</td>
<td>Required</td>
</tr>
<tr>
<td>HPSRAMSO</td>
<td>Output</td>
<td>LRECL=133</td>
<td>Optional</td>
</tr>
<tr>
<td>SNAP</td>
<td>Output</td>
<td>LRECL=133</td>
<td>Optional</td>
</tr>
<tr>
<td>SYSUDUMP</td>
<td>Output</td>
<td>LRECL=133</td>
<td>Optional</td>
</tr>
<tr>
<td>SORTWKnn</td>
<td>Input/Output</td>
<td></td>
<td>Optional</td>
</tr>
</tbody>
</table>
Table 2. DD statements and record format for IMS HP Load (continued)

<table>
<thead>
<tr>
<th>DDNAME</th>
<th>Use</th>
<th>Format</th>
<th>Need</th>
</tr>
</thead>
<tbody>
<tr>
<td>HPSRILEA</td>
<td>Input/Output</td>
<td></td>
<td>Optional</td>
</tr>
<tr>
<td>HPSRILEB</td>
<td>Input/Output</td>
<td></td>
<td>Optional</td>
</tr>
<tr>
<td>HPSRWknn</td>
<td>Input/Output</td>
<td></td>
<td>Optional</td>
</tr>
</tbody>
</table>

Notes:
1. When you specify HPIO=YES or when you delete and define database data sets by specifying the DDEFPDS DD statement, the load module library of STEPLIB must be authorized through the Authorized Program Facility (APF).
2. The large format data set can be specified when z/OS Version 1 Release 7 or later is used.
3. If you use dynamic allocation, do not use the DD statements for these data sets.
4. HPSRknn DD statements are effective only when the WFPHELQ control statement is specified.
5. This data set is required only if the SEQERROR=SAVE control statement is specified.

EXEC  Must be in the following form:
```
//EXEC EXEC PGM=HPSRRC00,PARM='ULU,DFSURGL0,dbname'
```

`dbname` is the name of the DBD that includes the database to be reloaded. The usual IMS positional parameters, such as DBRC and SWAP, can follow the `dbname`.

To reload a HALDB, you must set DBRC=Y.

If you do not use the functions that require authorization for the STEPLIB DD (for example, HPIO, COMPAUTH, and DDEFPDS), you can specify DFSRRC00 as follows:
```
//EXEC EXEC PGM=DFSRRC00,PARM='ULU,DFSURGL0,dbname'
```

To reload an IMS catalog database that is not registered in the RECON data set, either modify the EXEC statement or the STEPLIB DD statement as follows:

- Specify an additional EXEC parameter, DFSDF, as shown in the following example:
```
//EXEC EXEC PGM=HPSRRC00,PARM='ULU,DFSURGL0,DFSCD000,
 //           N,N,,,,,,,,,,,,'DFSDF=CAT')
```

Replace the 3-character suffix CAT with the suffix of the DFSDFxxx member that specifies the unregistered IMS catalog database name.

- Place the IMS Catalog Definition exit routine (DFS3CDX0) in the STEPLIB DD concatenation.

With either method, you must allocate the catalog partition definition data set. To do so, add the DFSHDBSC DD statement or the DFSMDA member with the TYPE=CATDBDEF statement.

STEPLIB DD  Points to two library data sets that must be in the following order:
```
//STEPLIB DD DISP=SHR,DSNAME=pgmlib
   DD DISP=SHR,DSNAME=IMSESA.RESLIB
```

`pgmlib` is the name of the library that contains the load module of IMS HP Load. IMSESA.RESLIB is the name of the library that contains the IMS load modules.
If you specify HPIO=YES, COMPAUTH=YES, or DDEFPDS DD, the load module libraries that are specified on the STEPLIB DD must be APF-authorized.

**IMSDLIB DD**
Specifies a partitioned data set or data sets that contain the dynamic allocation members for the database, index, and the RECON data sets.

Allocation of the database, index, and the RECON data sets are attempted in the following order:
1. The DD statements that are coded in the JCL stream
2. Dynamic allocation members in the IMSDALIB concatenation
3. Dynamic allocation members in JOBLIB or STEPLIB concatenation

For dynamic allocation of HALDB databases, the RECON data sets are used instead of the IMSDALIB DD statement.

**IMS DD**
Describes the library that contains the DBD referred to in the PARM field of the EXEC statement. This data set must reside on a direct-access device.

**PROCLIB DD**
Points to the IMS.PROCLIB data set.

If you specify DFSDF=xxx in the EXEC statement, you must also specify this DD statement and the DFSDFxxx member must exist in the IMS.PROCLIB data set.

**DFSRESLB DD**
Points to an authorized library that contains IMS SVC modules. For an IMS HP Load job, IMSESA.RESLIB and any data set that is concatenated to it on the DFSRESLB DD statement must be authorized through the Authorized Program Facility (APF). If you do not specify IMSESA.RESLIB in the STEPLIB DD statement or if a library that is concatenated to the STEPLIB DD statement is not APF-authorized, the DFSRESLB DD statement is needed.

For more information, see *IMS System Definition*.

**DFSUINPT DD**
Describes the input data set that contains the data to be reloaded. This data set must be created by the unload utilities of IMS High Performance Unload or an equivalent utility. The data set must reside on either a tape or a direct-access device.

**DFSURCDS DD**
Defines the control data set for this program. The data set must be the output that is generated by the IMS Database Prereorganization utility (DFSURPR0).

This DD statement is applicable only for logical relationships and must be included if logical relationships exist.

This data set must reside on either a tape or a direct-access device.

**DFSVSAMP DD**
Describes the data set that contains the buffer pool information that is required by the IMS DL/I buffer handler. For more information about the control statement format, see *IMS Database Administration*.

**Notes:**
• When the DFSVSAMP DD card is omitted, the DL/I Buffer Pool is not allocated.
• You must specify DFSVSAMP DD when reloading HALDB or when using the DL/I call in the user exit routine.

FRRIN DD
Defines the optional data set that contains control statements that specify the optional functions of IMS HP Load.

DDEFPDS DD
Defines the optional data set that contains the IDCAMS commands for deleting and defining the database data sets to be processed by IMS HP Load. IMS HP Load searches for the delete and define statements for each database data set in the member that has the same name as the database DD name.

The member must contain only the delete and define statements.

Requirements:
• To delete and define database data sets by specifying this DD statement, the load module library of STEPLIB must be authorized through the Authorized Program Facility (APF).
• Program HPSRRC00 must be specified on the EXEC statement.
• To delete and define HALDB data sets by specifying this DD statement, the input unloaded data set must be created by IMS HPUnload with the CHECKREC=YES option. The unloaded data set must not be concatenated.

RECON1 DD
Defines the first Data Base Recovery Control (DBRC) RECON data set.

RECON2 DD
Defines the second DBRC RECON data set.

RECON3 DD
Defines the optional DBRC RECON data set that is used when an error is encountered in RECON1 or RECON2. This RECON data set must be the same RECON data set as the control region is using.

Restriction: If you use dynamic allocation, do not use these RECON data set ddnames.

database DD
Defines the database data set to be reorganized. You need one statement for each data set that appears in the DBD that describes this database. The ddname must match the ddname in the DBD.

For an HIDAM database, DD statements must also exist for the data sets that represent the index. The DD statements that relate to the index must contain ddnames specified in the DBD for the index database. No DD statements are required for any secondary indexes that are associated with this database.

For a HISAM database, DD statements must exist for both the primary data set (KSDS) and the overflow (ESDS) data sets. The DD name for primary and the overflow data sets must be the same as the ddname in the DBD that was referred to when this data set was unloaded.

This data set must reside on a direct-access device.
Recommendation: For VSAM database data sets, the SPEED option for DEFINE CLUSTER is strongly recommended. The SPEED option improves the performance of IMS HP Load.

IMS HP Load optimizes the buffer specification for the database data set by itself. You do not need to specify the AMP= parameter or the DCB=(NCP=) parameter for the DD statement.

Restriction: If you use dynamic allocation, do not use the DD statement for the database data sets.

For HALDBs, no database DD statements are necessary for any database data sets because the data sets are always dynamically allocated.

**DFSURWF1 DD**

Describes the work data set to be created during reload.

The data set is used as input to the following utilities to resolve logical or secondary index relationships:

- IMS Index Builder
- IMS High Performance Prefix Resolution
- IMS Database Prefix Resolution utility

This DD statement is necessary if at least one of the following conditions are met:

- The database is HDAM, HIDAM, or HISAM and it has logical relationship.
- The database has a secondary index and INDEXBLD=NO is specified for the IPR Driver, but the HPSRSIDX DD statement is not coded.

**Notes:**

- If WF1REC=NO is specified, you can omit this DD statement. If the preceding condition is not met, this DD statement is not required.
- If you omit the DFSURWF1 DD statement and specify the WF1DSN statement, IMS HP Load allocates the DFSURWF1 data set dynamically. You can specify the attributes of the data set by using WF1SPACE, WF1UNIT, and WF1VOLUME statements.

If HPSRSIDX DD is specified, only records of logical relationship are written to this data set, and it can be used as an input for utilities to resolve logical relationships.

You can specify DUMMY if the database being reloaded is neither involved in a logical relationship nor with a secondary index.

The DCB parameters for the DD statement must include RECFM=VB and BLKSIZE. The specification must be the same as that for the work data set of the user’s initial load program or for the Database Scan utility (DFSURGS0). A value of LRECL=900 is the default and recommended, but a smaller value (as small as 300) can be used if no secondary indexes are present or if HPSRSIDX DD is specified.

The data set must reside on either a tape or a direct-access device.

**HPSRPnnn DD**

Describes the WFP work data sets. WFP data sets are work data sets that can be used by Index Builder to rebuild partition secondary indexes (PSINDEXes). These data sets are created while IMS HP Load reloads HALDB partitions. One work data set is created for each HALDB partition.
If you want IMS HP Load to dynamically allocate WFP data sets, omit this DD statement.

For \textit{nnn}, specify a 3-digit decimal number from 001 to 999. The value can also be T00 or T01. T01 represents 1000, and T01 represents 1001. If you designate IMS HP Load to dynamically allocate the WFP data sets, IMS HP Load uses sequence numbers for \textit{nnn}.

The data sets must reside on a direct-access device.

Create WFP data sets when all of the following conditions are met:

- The database has PSINDEXes.
- You are going to rebuild the PSINDEXes by using IMS Index Builder in a subsequent job step or in another job.
- You want to avoid scanning the database while rebuilding indexes with IMS Index Builder.

If you are not sure whether the HALDB database contains empty partitions, use the unloaded data set that is created by IMS HP Unload with the CHECKREC=YES option. Otherwise, WFP data sets are not created for the empty partitions.

**Naming convention for WFP data sets**

If you specify the names of WFP data sets by using HPSRP\textit{nnn} DD statements, ensure that the data set names on the DSN parameters conform to the following naming convention.

The prefix for the WFP data sets is specified by the WFPHLQ control statement. For each HPSRP\textit{nnn} DD statement specified, specify the data set name as follows:

\begin{equation*}
\text{DSN} = \text{WFPHLQ	extunderscore value	extunderscore partition	extunderscore name}.WFP
\end{equation*}

For example, if you specify WFPHLQ=IMSTOOLS.WORK.HMPORDP1 to define the prefix, and if you specify HPSRP001 and HPSRP002 DD statements for partitions that are named PART001 and PART002, the DSN parameters for the WFP data sets must be specified as follows:

\begin{align*}
\text{DSN} &= \text{IMSTOOLS	extunderscore WORK	extunderscore HMPORDP1	extunderscore PART001	extunderscore WFP} \\
\text{DSN} &= \text{IMSTOOLS	extunderscore WORK	extunderscore HMPORDP1	extunderscore PART002	extunderscore WFP}
\end{align*}

If you do not specify this statement, IMS HP Load allocates the data sets dynamically by using the values summarized in the following table.

**Requirement:** To allocate the data sets dynamically, use z/OS V1.7 or higher.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DDN</td>
<td>HPSRP\textit{nnn}</td>
<td>\textit{nnn}: Sequence number</td>
</tr>
</tbody>
</table>
| DSN       | hlq.partition\_name.WFP | hlq: The value that is specified on the WFPHLQ control statement  
 partition\_name: The partition name of the primary database  
 WFP: Fixed value |
| DISP      | NEW,CATLG,DELETE |                                                |

Table 3. \textit{DYNAALLOC} parameters for dynamically allocating WFP data sets
Table 3. DYNALLOC parameters for dynamically allocating WFP data sets (continued)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UNIT</td>
<td>unit_name</td>
<td>The value that is specified on the WFPUNIT control statement. If the control statement is not specified, SYSALLDA is used.</td>
</tr>
<tr>
<td>SPACE</td>
<td>![TRK, or CYL, primary_alloc, secondary_alloc, RLSE]</td>
<td>The value that is specified on the WFPSPACE control statement. If the control statement is not specified, the region size is estimated based on the allocation size of the partition data set.</td>
</tr>
<tr>
<td>RECFM</td>
<td>FB</td>
<td></td>
</tr>
<tr>
<td>LRECL</td>
<td>7250</td>
<td></td>
</tr>
<tr>
<td>VOL=SER</td>
<td>![volume_name or *]</td>
<td>The value that is specified on the WFPVOL control statement. If the control statement is not specified, the VOLUME parameter for the DYNALLOC macro is not set.</td>
</tr>
<tr>
<td>DSNTYPE</td>
<td>LARGE</td>
<td></td>
</tr>
</tbody>
</table>

When IMS HP Load attempts to dynamically allocate a WFP data set and if the WFP data set already exists with the intended data set name, IMS HP Load deletes the existing data set and creates the data set with the same name.

**HPSRSIDX DD**

Defines the optional output work data set. If it is specified and secondary indexes are present in the DBDs being processed, the data set is used as input to utilities to resolve secondary index relationships. The use of this data set increases the performance of utilities to resolve secondary index relationships because it has fewer records to read. DCB parameters specified within this program are RECFM=VB and LRECL=900. BLKSIZE must be provided on the HPSRSIDX DD statement. The data set must reside on either a tape or a direct-access device. This DD statement is not necessary if the database has no secondary index or INDEXBLD=YES is specified for the IPR Driver.

**OVERFLOW DD**

Defines the optional sequential data set that is used as a temporary storage to keep database blocks that are directed to the overflow area during HDAM RAA reloading. Such blocks are written into the overflow area after RAA reloading is completed. This data set is used only when both OADSPR=Y and HPIO=N are specified. The SPACE parameter for the DD statement must be large enough to hold all CIs or blocks in the overflow area. The DCB parameters must include RECFM=VB. This data set must reside on either a tape or a direct-access device. If both OADSPR=Y and HPIO=N are specified and this DD statement is not specified, IMS HP Load attempts to allocate dynamically on a direct-access device the value calculated internally according to the following method.

**Method of calculating size for dynamic allocation:** The number of database blocks directed to the overflow area during HDAM RAA reloading cannot be estimated in advance. For optimization, IMS HP Load calculates the cylinders for OVERFLOW DD, which can be 25% of RAA cylinders. IMS HP Load uses the same value to calculate both primary and secondary values. If the calculated value is less than 10 cylinders, 10 cylinders are used.
The way to maintain the blocks in OVERFLOW DD is different from the one in an actual database data set, so the number of cylinders for the OVERFLOW DD is not directly comparable with the 25% of RAA cylinders.

For calculation, IMS HP Load uses the 3390 device type geometry regardless of the actual device type of OVERFLOW DD, because the device type cannot be known prior to allocation. If a 3380 or a 9345 DASD is used, for example, the effective percentage of RAA of the allocation amount is a little smaller, and will be approximately 20%.

However, there is no formula to estimate the space requirement for OVERFLOW DD beforehand. The size of OVERFLOW DD depends on the number of segments that is to be directed to the overflow part of the database. But the size equal to the overflow part of the original database is enough, until you change the DBD definition or the randomizer.

**HPSRISMG DD**

This optional output data set contains the messages that are issued by DFSORT. This data set is used only when ILDSBLD=YES,DSPR is specified. If the DD statement is omitted, the data set is allocated dynamically.

**SEQERROR DD**

Defines a sequential output data set in which segment data in key sequence error is to be kept. This data set is required only if SEQERROR=SAVE is specified. This data set saves segments in key sequence error and their hierarchical dependent segments. If the DCB parameter is specified on the DD, the record format VB (RECFM=VB) is required. Do not specify the logical record length or the block size of the data set.

**SYSPRINT DD**

Defines the statistics output data set. The data set can reside on a tape or a direct-access device, or it can be routed through the output stream.

**HPSRAMSO DD**

Defines the output data set for the Result of Database Data Set Allocation report. The data set can reside on a tape, a direct-access device, or it can be routed through the output stream. If this DD statement is not coded, the data set is dynamically allocated to the SYSOUT stream of the output class that is specified by the MSGCLASS parameter on the JOB statement.

**SNAP DD**

Defines a snap dump data set. It contains printed dumps of the DBD for your reference. The snap dump is activated by the DBDSNAP=YES specification in the FRRIN data set. If no snap dump is required, the DUMMY statement can be specified.

**SYSUDUMP DD**

Defines a dump data set. Most errors encountered end the IMS HP Load by issuing an ABEND macro with the DUMP operand. Only a SYSUDUMP DD is needed. The SYSABEND DD statement is not required.

**SORTWKnnn DD**

These data sets are intermediate storage data sets used by DFSORT. If SORT=YES or SORTHIDAM=YES is specified, you can define these data sets and adjust the size of work space that is used for the sort process. You can specify multiple data sets, 'nnn' showing its serial number. This
specification is optional. If you do not specify the statement, the SORTWKn DD statements, these data sets are dynamically allocated by DFSORT.

- For a (P)HDAM database, the size is estimated based on the size of RAA.
- For a HIDAM database, the size is estimated based on the primary allocation size of the primary data set group of the HIDAM database.
- For a PHIDAM database, the size is estimated based on the number of partitions.

If the number of unloaded records is provided by the RECNUM statement, this value supersedes other values and will be used for estimation.

If the estimated size is extremely large or extremely small, specify SORTWKn DD explicitly.

For more information about how to code SORTWKn DD statements, see the DFSORT Application Programming Guide.

**HPSRILEA/HPSRILEB DD**

These data sets are used by DFSORT, as substitute for SORTIN data sets, when ILDSBLD=YES,DSPR is specified. If you do not specify the statement, the data sets are allocated dynamically with sizes estimated on the basis of the allocation size of ILDS.

To avoid space shortage problems that might be caused by extension of the data sets, the size estimation is adjusted as follows:

- If the high used RBA of ILDS is greater than the primary space amount, the equivalent amount is specified as the primary space amount.
- If the secondary space amount is less than 1/5 of the high used RBA of ILDS, the equivalent amount is specified as the secondary space amount.
- If the primary or secondary space amount of ILDS is less than 100 cylinders, 100 cylinder is used as the space amount.
- The data set is allocated with volume count so that it can be extended on up to 10 volumes.

You can specify the data class and the storage class for HPSRILEA and HPSRILEB data sets by using ILEDATA and ILESTORC control statements.

A dummy data set or the DISP=MOD parameter cannot be specified on the HPSRILEA/HPSRILEB DD statement.

For more information about SORTIN data sets, see the DFSORT Application Programming Guide.

**HPSRWKnn DD**

These data sets are used by DFSORT, as substitute for SORTWK data sets, when ILDSBLD=YES,DSPR is specified. Here, nn shows its serial number. If you do not specify the statement, the data sets are allocated dynamically by DFSORT.

For more information, see the DFSORT Application Programming Guide.

**Notes:**

- The DFSCTL data set is not supported.
- Any DD name that starts with SYS, DFS, HPS, or FAB is reserved for system use.
Default values for the data set access method buffer

IMS HP Load uses a performance default of the access method buffer number for each input and output data set. The following table lists the default values.

Table 4. IMS HP Load defaults for the data set access method buffer

<table>
<thead>
<tr>
<th>DDNAME</th>
<th>Use</th>
<th>Access</th>
<th>Default parameter</th>
<th>Storage used</th>
</tr>
</thead>
<tbody>
<tr>
<td>DFSUINPT</td>
<td>In</td>
<td>QSAM</td>
<td>DCB=(BUFNO=60)</td>
<td>private high</td>
</tr>
<tr>
<td>HPSRSIDX</td>
<td>Out</td>
<td>QSAM</td>
<td>DCB=(BUFNO=40)</td>
<td>private low</td>
</tr>
<tr>
<td>DFSURWF1</td>
<td>Out</td>
<td>QSAM</td>
<td>DCB=(BUFNO=40)</td>
<td>private low</td>
</tr>
<tr>
<td>HPSRPnnn</td>
<td>Out</td>
<td>QSAM</td>
<td>DCB=(BUFNO=105)</td>
<td>private high</td>
</tr>
<tr>
<td>database</td>
<td>Out</td>
<td>ESDS</td>
<td>DBIOBUF=30</td>
<td>private high</td>
</tr>
<tr>
<td>database</td>
<td>Out</td>
<td>BSAM</td>
<td>DBIOBUF=15</td>
<td>private high</td>
</tr>
</tbody>
</table>

Notes:

1. IMS HP Load calculates the number of VSAM buffers that fits into 30 tracks (that is, the default value of the DBIOBUF parameter) depending on the CI size, and specifies AMP=('BUFND='nn') in the ESDS database data set internally.

2. IMS HP Load calculates the number of VSAM buffers that fits into 30 tracks depending on the CI size, and specifies AMP=('BUFND='nn,BUFNI=100') in the KSDS database data set internally. BUFNI=100 is used commonly for any CI size.

3. IMS HP Load calculates the number of BSAM buffers that fits into 15 tracks (that is, the default value of the DBIOBUF parameter) depending on the block size, and specifies DCB=(NCP='nn') in the OSAM database data set internally.
Control statements

Optional IMS HP Load control statements can be specified on the FRRIN data set.

Tip: Some of the control statement keywords and their values can be set as the default options for your runtime environment. For details about setting the default options, see Chapter 17, “Setting installation default options,” on page 241.

Overview of control statements

IMS HP Load supports various options to control the behavior of IMS HP Load jobs.

Subsections:

• “Control statements for process control”
• “Control statement for performance” on page 47
• “Control statements for reloading a HALDB database” on page 48
• “Control statements for performance diagnosis” on page 48
• “Control statement syntax” on page 49

Control statements for process control

The following table lists the keywords that you can specify for the process control of the FRRIN data set DD statement.

Table 5. IMS HP Load control statements for process control

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted values</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMSETDSG</td>
<td>n or (n1, ..., n10)</td>
<td>*</td>
<td>Specifies the threshold value that is used to set the bitmap block. See “BMSETDSG control statement” on page 49.</td>
</tr>
<tr>
<td></td>
<td>n=value or *</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHKEMPTYUL</td>
<td>YES, WARN, n</td>
<td>YES,WARN,4</td>
<td>Specifies whether to check if the unloaded data set contains segment data. See “CHKEMPTYUL control statement” on page 50.</td>
</tr>
<tr>
<td></td>
<td>NO</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>CHKTRAIL</td>
<td>YES, WARN, ERROR</td>
<td>YES,WARN</td>
<td>Specifies whether to check for the existence of the trailer record. See “CHKTRAIL control statement” on page 51.</td>
</tr>
<tr>
<td></td>
<td>NO</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>CHKUPDT</td>
<td>YES, ABEND</td>
<td>NO</td>
<td>Specifies whether the check for an update is done between unload and reload processes. See “CHKUPDT control statement” on page 52.</td>
</tr>
<tr>
<td></td>
<td>YES, WARN</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NO</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>COMPAUTH</td>
<td>YES or NO</td>
<td>NO</td>
<td>Specifies whether to call the segment compression exit in supervisor state. See “COMPAUTH control statement” on page 52.</td>
</tr>
<tr>
<td>COMPRESS</td>
<td>YES or NO</td>
<td>YES</td>
<td>Compresses segments during reload. See “COMPRESS control statement” on page 53.</td>
</tr>
<tr>
<td>DATXEXIT</td>
<td>YES or NO</td>
<td>NO</td>
<td>Controls the IMS/ESA Year 2000 Exit Tool support. See “DATXEXIT control statement” on page 54.</td>
</tr>
<tr>
<td>EXTENT</td>
<td>1–60</td>
<td>52</td>
<td>Specifies the maximum number of extensions of the OSAM data set that can be created. See “EXTENT control statement” on page 56.</td>
</tr>
<tr>
<td>Keyword</td>
<td>Accepted values</td>
<td>Default</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>-----------------</td>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>FRSPC</td>
<td>YES or NO</td>
<td>YES</td>
<td>Secures free space factors. See “FRSPC control statement” on page 57.</td>
</tr>
<tr>
<td>GDGBASE</td>
<td>YES or NO</td>
<td>NO</td>
<td>Specifies whether to create a GDG base entry. See “GDGBASE control statement” on page 57.</td>
</tr>
<tr>
<td>GDGLIMIT</td>
<td>aaa</td>
<td>10</td>
<td>Specifies the maximum number of GDS that can be associated with the GDG that is to be created. See “GDGLIMIT control statement” on page 57.</td>
</tr>
<tr>
<td>HPIO</td>
<td>YES or NO</td>
<td>YES</td>
<td>Specifies whether IMS HP Load uses the High Performance I/O Driver for output data. See “HPIO control statement” on page 58.</td>
</tr>
<tr>
<td>INPUT</td>
<td>DFSUINPT, exitname,L</td>
<td>DFSUINPT</td>
<td>Specifies the input data set, the user exit routine, and its language. See “INPUT control statement” on page 60.</td>
</tr>
<tr>
<td>ITKBLOAD</td>
<td>library_name or *NO</td>
<td>*NO</td>
<td>Specifies the library name of IMS Tools KB. See “ITKBLOAD control statement” on page 61.</td>
</tr>
<tr>
<td>ITKBSRVR</td>
<td>server_name or *NO</td>
<td>*NO</td>
<td>Specifies the name of the IMS Tools KB server. See “ITKBSRVR control statement” on page 61.</td>
</tr>
<tr>
<td>OADSPR</td>
<td>YES or NO</td>
<td>NO</td>
<td>Specifies whether to reduce the usage of data space to process (P)HDAM OA. See “OADSPR control statement” on page 63.</td>
</tr>
<tr>
<td>PAD</td>
<td>X'00'-X'FF'</td>
<td>X'00'</td>
<td>Specifies the padding character to fill out the segments to adjust to the length defined in the DBD definition. See “PAD control statement” on page 64.</td>
</tr>
<tr>
<td>PROGMON</td>
<td>0–999999</td>
<td>0</td>
<td>Monitors job progress. See “PROGMON control statement” on page 66.</td>
</tr>
<tr>
<td>RAAFORMAT</td>
<td>YES or NO</td>
<td>YES</td>
<td>Formats the HDAM RAA. See “RAAFORMAT control statement” on page 66.</td>
</tr>
<tr>
<td>RECNUM</td>
<td>mmmmmnnnnn</td>
<td>(none)</td>
<td>Specifies the number of unload records that are to be reloaded. See “RECNUM control statement” on page 67.</td>
</tr>
<tr>
<td>RMBSEGSZ</td>
<td>YES or NO</td>
<td>NO</td>
<td>Specifies how IMS HP Load calculates the database record size to determine whether to put the records into the overflow area. See “RMBSEGSZ control statement” on page 68.</td>
</tr>
<tr>
<td>SEARCH</td>
<td>0–999 -999–0</td>
<td>3</td>
<td>Sets the space search boundary. See “SEARCH control statement” on page 68.</td>
</tr>
<tr>
<td>SEQERROR</td>
<td>ABEND or SAVE</td>
<td>ABEND</td>
<td>Handles key sequence errors. See “SEQERROR control statement” on page 69.</td>
</tr>
<tr>
<td>SORT</td>
<td>YES, NO, or COND</td>
<td>NO</td>
<td>Specifies whether to run physical sequential sort before loading with (P)HDAM. See “SORT control statement” on page 70.</td>
</tr>
<tr>
<td>SORTHIDAM</td>
<td>YES or NO</td>
<td>NO</td>
<td>Specifies whether to run the sort processing before loading a HIDAM or PHIDAM. See “SORTHIDAM control statement” on page 72.</td>
</tr>
<tr>
<td>TWINSTAT</td>
<td>YES or NO</td>
<td>NO</td>
<td>Prints statistics about the twin chained segments in the Segment Statistics report. See “TWINSTAT control statement” on page 72.</td>
</tr>
</tbody>
</table>
Table 5. IMS HP Load control statements for process control (continued)

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted values</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>USEREXIT</td>
<td>exitname</td>
<td>(none)</td>
<td>Specifies the name of the user exit routine. See “USEREXIT control statement” on page 73.</td>
</tr>
<tr>
<td>USERHDR</td>
<td>element</td>
<td>(none)</td>
<td>Specifies the name of the element of the header portion of the unloaded record. See “USERHDR control statement” on page 73.</td>
</tr>
<tr>
<td>USERHDRX</td>
<td>exitname</td>
<td>(none)</td>
<td>Specifies the name of the user header conversion exit routine. See “USERHDRX control statement” on page 75.</td>
</tr>
<tr>
<td>WF1DSN</td>
<td>data_set_name or GDG_base_name.&amp;GDG</td>
<td>(none)</td>
<td>Specifies the data set name for the DFSURWF1 data set when dynamically allocating the data set. See “WF1DSN control statement” on page 75.</td>
</tr>
<tr>
<td>WF1REC</td>
<td>YES or NO</td>
<td>YES</td>
<td>Specifies whether to create WF1 records that are written to the DFSURWF1 output work data set. See “WF1REC control statement” on page 76.</td>
</tr>
<tr>
<td>WF1SPACE</td>
<td>CYL or TRK, primary_alloc, secondary_alloc</td>
<td>(none)</td>
<td>Specifies the space parameter for the DFSURWF1 data set when dynamically allocating the data set. See “WF1SPACE control statement” on page 76.</td>
</tr>
<tr>
<td>WF1UNIT</td>
<td>unit_name</td>
<td>SYSALLDA</td>
<td>Specifies the device for the DFSURWF1 data set when dynamically allocating the data set. See “WF1UNIT control statement” on page 77.</td>
</tr>
<tr>
<td>WF1VOL</td>
<td>volume_name or *</td>
<td>(none)</td>
<td>Specifies the volume names for the DFSURWF1 data set when dynamically allocating the data set. See “WF1VOL control statement” on page 77.</td>
</tr>
<tr>
<td>WFPHLQ</td>
<td>hlq</td>
<td>(none)</td>
<td>Specifies the high-level qualifier for the WFP data sets. See “WFPHLQ control statement” on page 78.</td>
</tr>
<tr>
<td>WFPSPACE</td>
<td>CYL or TRK, primary_alloc, secondary_alloc</td>
<td>(none)</td>
<td>Specifies the SPACE parameter for the WFP data sets when allocating WFP data sets dynamically. See “WFPSPACE control statement” on page 78.</td>
</tr>
<tr>
<td>WFPUNIT</td>
<td>unit_name</td>
<td>SYSALLDA</td>
<td>Specifies the device for the WFP data sets when allocating WFP data sets dynamically. See “WFPUNIT control statement” on page 79.</td>
</tr>
<tr>
<td>WFPVOL</td>
<td>volume_name or *</td>
<td>(none)</td>
<td>Specifies the volume names for the WFP data sets when allocating WFP data sets dynamically. See “WFPVOL control statement” on page 79.</td>
</tr>
<tr>
<td>WFPVOLCNT</td>
<td>nnn</td>
<td>(none)</td>
<td>Specifies the volume count for the WFP data sets when allocating WFP data sets dynamically. See “WFPVOLCNT control statement” on page 80.</td>
</tr>
</tbody>
</table>

Control statement for performance

The following table lists the keywords you can specify for performance improvement.
### Table 6. IMS HP Load control statements for performance

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted values</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BufferNo</td>
<td>5–32767</td>
<td>N/A</td>
<td>Sets the number of IMS HP Load buffers. See “BUFFERNO control statement” on page 50.</td>
</tr>
<tr>
<td>DBIOBUF</td>
<td>1–999</td>
<td>30 (for VSAM) 15 (for OSAM)</td>
<td>See Note 2</td>
</tr>
<tr>
<td>DBRLBUF</td>
<td>1–999</td>
<td>45</td>
<td>Sets the number of IMS HP Load buffers. See “DBRLBUF control statement” on page 56.</td>
</tr>
</tbody>
</table>

**Notes:**
1. DBRLBUF and BUFFERNO are mutually exclusive.
2. For HISAM, the default value for DBIOBUF is 15, and the default value for DBRLBUF is 2.

#### Control statements for reloading a HALDB database

The following table lists the keywords you can specify when reloading a HALDB database.

### Table 7. IMS HP Load control statements for reloading a HALDB database

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted values</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ILDSBLD</td>
<td>C, YES, DSPR, NO</td>
<td>C</td>
<td>Specifies whether to build ILDS and how to build it during reorganization. See “ILDSBLD control statement” on page 58.</td>
</tr>
<tr>
<td>ILEDATA</td>
<td>data_class_name</td>
<td>(none)</td>
<td>Specifies the data class (DATACLASS) for HPSRILEA and HPSRILEB. See “ILEDATA control statement” on page 59.</td>
</tr>
<tr>
<td>ILESTORC</td>
<td>storage_class_name</td>
<td>(none)</td>
<td>Specifies the storage class (STORCLASS) for HPSRILEA and HPSRILEB. See “ILESTORC control statement” on page 60.</td>
</tr>
<tr>
<td>LPART</td>
<td>(partname,nnn), (part1,...,partn), or *ALL</td>
<td>N/A</td>
<td>Specifies the HALDB partitions to load. See “LPART control statement” on page 62.</td>
</tr>
<tr>
<td>PARTINIT</td>
<td>NO, YES, PINITON, UNLOAD, LPART</td>
<td>NO</td>
<td>Initializes partitions not populated by any segment during reload. See “PARTINIT control statement” on page 65.</td>
</tr>
<tr>
<td>PARTSTAT</td>
<td>YES or NO</td>
<td>YES</td>
<td>Generates the statistics report for the partition. See “PARTSTAT control statement” on page 66.</td>
</tr>
<tr>
<td>REPAIRILK</td>
<td>NO or YES</td>
<td>NO</td>
<td>Specifies whether to reassign indirect list keys (ILKs). See “REPAIRILK control statement” on page 67.</td>
</tr>
</tbody>
</table>

#### Control statements for performance diagnosis

The following table lists the keywords that you can specify for performance diagnosis.
Table 8. IMS HP Load control statements for performance diagnosis

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted values</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBDSNAP</td>
<td>YES or NO</td>
<td>NO</td>
<td>Generates the DBD Snap report. See “DBDSNAP control statement” on page 55.</td>
</tr>
</tbody>
</table>

Control statement syntax

A keyword and its associated value can be coded in a free format (columns 1–72) provided these syntactical coding rules are followed:

- A keyword and its value must be on a single control statement.
- A keyword must be separated from its associated value by an equal sign (=). The equal sign must not be preceded by blanks, and not be followed by blanks; that is, a null value is not allowed.
- A comment line must begin with an asterisk (*) in column 1.
- You can abbreviate YES as Y and NO as N.

BMSETDSG control statement

Specifies the threshold value that is used to set the bitmap block of an HDAM, an HIDAM, a PHDAM, or a PHIDAM database. The value can be specified for each data set group (DSG).

This control statement is optional. If you do not specify this control statement, BMSETDSG=* is used.

```
BMSETDSG=(value)
```

* Specifies to use the length of the longest segment in the data set group in the same manner as IMS.

value Specifies to use a threshold value from 3 to 32,000 for the data set group.

You can specify one parameter or a list of parameters. If one parameter is specified, it will be used for the first data set group of the database. Otherwise, parameters must be separated by commas and enclosed in parentheses. The Nth parameter will be used for the Nth data set group. You can specify up to 10 parameters. If the number of values specified is less than the number of data set groups, '*' will be used for other data set groups for which values are not specified.

Example 1

```
BMSETDSG=1000
```

The threshold value for the first DSG of the database is 1000.

Example 2

```
BMSETDSG=(1000, *, 1200)
```
The threshold value for the first DSG is 1000 and the value for the third DSG is 1200. For other DSGs, the length of the longest segment in the data set group will be used.

**BUFFERNO control statement**

Specifies the number of database buffers that are used for one database data set group when reading or writing the database.

This statement is mutually exclusive with the DBRLBUF statement. Determine whether you use DBRLBUF to specify the amount of database buffers or use BUFFERNO to specify the number of database buffers.

The size of a database buffer is almost the same as the CI or block size of the database data set group. Because the database buffers are acquired by the data set group, the total number of database buffers is the sum of database buffers in each data set group.

This control statement is optional. If you do not specify this control statement, the DBRLBUF specification is used. If neither control statement is specified, DBRLBUF=45 is used.

```
BUFFerno=nnnnn
```

The abbreviation BUFNO= can be used for BUFFERNO=.

```
nnnnn  Specifies a left-aligned decimal number in the range of 5 - 32767. There is no default.
```

**Note:** You can reduce EXCPs by increasing the number of database buffers. The amount of database buffers must be greater than the amount of access method buffers plus 10 tracks. If the amount of database buffers is insufficient, IMS HP Load internally adjusts the amount of database buffers. Select the suitable value for the number of database buffers based on your execution environment.

**CHKEMPTYUL control statement**

Specifies whether to check if the unloaded data set contains segment data. Also, specifies the return code that is set when no segment data is found in the unloaded data set.

If DUMMY is specified for the DFSUINPT DD statement, this check is not applicable.

This control statement is optional. If you do not specify this control statement, CHKEMPTYUL=YES,WARN,4 is used.
YES  Specifies to check if segment data exists in the unloaded data set.

WARN,\( n \)
Sets the job step return code to \( n \) when the unloaded data set contains no segment data. For \( n \), specify a left-aligned decimal number in the range of 1 - 4.

If you specify CHKEMPTYUL=YES or CHKEMPTYUL=YES,WARN, CHKEMPTYUL=YES,WARN,4 is used.

NO  Specifies not to check if segment data exists in the unloaded data set. The return code is not changed even if no segment data exists in the unloaded data set.

The return code that you specify with CHKEMPTYUL can be overridden by a higher return code set for some other reason.

Tip: The default value of the CHKEMPTYUL control statement can be changed. For details, see Chapter 17, “Setting installation default options,” on page 241.

**CHKTRAIL control statement**

Specifies whether to check if a trailer record exists in the *HD or *SH format of the unload data set.

This control statement is optional. If you do not specify this control statement, CHKTRAIL=YES,WARN is used.

YES  Specifies that the existence of the trailer record is checked.

WARN
Specifies IMS HP Load to continue processing, setting the job step return code to 4 when a trailer record is not found in the unload data set.

If you specify only CHKTRAIL=YES, CHKTRAIL=YES,WARN is assumed.

ERROR
Specifies IMS HP Load to end with return code 16 when a trailer record is not found in the unload data set.
NO Specifies that the existence of trailer record is not checked. IMS HP Load will continue processing even if a trailer record does not exist in the unload data set.

Tip: The default value of the CHKTRAIL control statement can be changed. For details, see Chapter 17, “Setting installation default options,” on page 241.

CHKUPDT control statement

Specifies whether you want to do the IMS HP Load check against DBRC to see if an update of the database has been done between unload and reload processes.

This control statement is optional. If you do not specify this control statement, CHKUPDT=NO is used.

NO Specifies that update is not checked.
YES Specifies that update between unload and reload processes is checked.
ABEND Specifies that IMS HP Load ends abnormally when an update is found.
    If you specify only CHKUPDT=YES, CHKUPDT=YES,ABEND is assumed.
WARN Specifies that IMS HP Load continues processing and sets the job step return code to 4 when an update is found.

Restriction: The CHKUPDT control statement is not valid when you use the Load API.

The check against DBRC for an update is activated when all of the following conditions are met:
• DBRC is active
• The database and database data sets are registered to DBRC
• The unloaded data set was created by the IMS HP Unload FABHURG1 utility with the CHECKREC YES option

Tip: The default value of the CHKUPDT control statement can be changed. For details, see Chapter 17, “Setting installation default options,” on page 241.

COMP AUTH control statement

Specifies whether to call the segment compression exit in supervisor state.

Tip: If you use an encryption exit of InfoSphere® Guardium® Data Encryption for DB2® and IMS Databases as the segment compression exit, specifying COMP AUTH=YES reduces performance degradation. Even if the input data for
IMS HP Load is created in compressed (encrypted) format, it is recommended to
specify COMPAUTH=YES to reduce performance degradation because IMS HP
Load calls the encryption exit to decrypt segment data for internal processes in
some conditions.

This control statement is optional. If you do not specify this control statement,
COMPAUTH=NO is used.

NO  Specifies to call the segment compression exit in problem state.
YES  Specifies to call the segment compression exit in supervisor state.

Notes:
• You must run this statement under program HPSRRC00, which is the IMS HP
  Load version of the region controller program.
• Under IMS region controller (PGM=DFSRRC00), COMPAUTH=NO is assumed.
  If you specify COMPAUTH=YES, it will be ignored.
• If you specify COMPAUTH=YES, you must APF-authorize the load module
  libraries that are specified on the STEPLIB DD.
• If one or more libraries on the STEPLIB DD are not APF-authorized, IMS HP
  Load uses COMPAUTH=NO even if you specify COMPAUTH=YES.

Tip: The default value of the COMPAUTH control statement can be changed. For
details, see Chapter 17, “Setting installation default options,” on page 241.

COMPRESS control statement
Specifies whether to use a segment compression exit. This statement activates or
deactivates calling the segment compress routine for all segment types with
COMPRTN specified on their related SEGM statements in the DBD.

This control statement is optional. If you do not specify this control statement,
COMPRESS=YES is used.

YES  Specifies to compress the segments when the unloaded data set is
decompressed.
NO   Specifies to process the segments as is when the unloaded data set is
      compressed.
If you use the output from the unload utilities of IMS High Performance Unload as an input, you do not need to specify the COMPRESS= keyword because IMS HP Load can automatically determine whether the segments are in the compressed format. To avoid specification errors, it is recommended that you do not specify this parameter.

The COMPRESS=NO parameter is required under the following conditions:
- If the input data set is created, in compressed format, by a utility other than IMS High Performance Unload or DBT V2R3 HSSR with APAR PQ22654.
- If the input data for the Load API function is created in compressed format.

Notes:
- An unloaded data set that is created by the unload utilities of IMS High Performance Unload with the DECOMPRESS=NO option cannot be used as an input for the IMS HD Reorganization Reload utility. If you use it, the HD reload job issues message DFS388A and ends with return code 16.
- An unloaded data set that is created by the unload utilities of IMS High Performance Unload with the DECOMPRESS=NO option cannot be used if you want to change the size of the segment by reorganization.

This control statement is in effect for all the database segments with COMPRTN= specified on their related SEGM statements in the DBD. This option cannot be selectively activated or deactivated for some compressed segments.

**DATXEXIT control statement**

Specifies whether to activate the IMS/ESA Year 2000 Exit Tool (5697-E04) support.

This control statement is optional. If you do not specify this control statement, DATXEXIT=NO is used.

![DATXEXIT control statement](image)

- **YES** Activates the IMS/ESA Year 2000 Exit Tool support. IMS HP Load treats the IMS/ESA Year 2000 Exit Tool in the same way that IMS does.
- **NO** Deactivates the IMS/ESA Year 2000 Exit Tool support. IMS HP Load does not call the IMS/ESA Year 2000 Exit Tool exit routine (the DFSDBUX1 exit routine) regardless of the DATXEXIT parameter in the DBD.

Use this option in conjunction with the DATXEXIT=NO option of the unload utility job of IMS High Performance Unload, which creates an unloaded data set in the stored form.

If DATXEXIT=YES is specified in the FRRIN data set, IMS HP Load calls the DFSDBUX1 exit routine if it exists in the STEPLIB. As the result of the first call of the DFSDBUX1 exit routine, IMS HP Load determines which segment type in the database needs to be converted. After that, IMS HP Load does not call the DFSDBUX1 exit routine for any segment type determined not to be converted.
When the segment types determined to be converted are unloaded in the compressed format by an unload utility of IMS High Performance Unload with the DECOMPRESS=NO option, IMS HP Load calls the segment edit/compression exit routine to decompress such segments before calling the DFSDBUX1 exit routine, and calls it again after the DFSDBUX1 exit routine to recompress the segments. Decompression and recompression are performed even if COMPRESS=NO is specified in the FRRIN data set.

If the segment data requires the DFSDBUX1 exit routine to be called and if a user exit routine is specified on the INPUT or USEREXIT control statement, IMS HP Load calls the user exit routine to edit or delete the segment data first, and then calls the DFSDBUX1 exit routine to convert the segment data.

If DATXEXIT=NO is specified in the FRRIN data set, IMS HP Load does not call the DFSDBUX1 exit routine even if it is in the STEPLIB. A user exit routine, if used, receives segment data not converted (in the stored form).

Note: If you use SEQERROR=SAVE in conjunction with DATXEXIT=NO, the SEQERROR data set will contain segments in the stored form because IMS HP Load does not deconvert such segments. You cannot use segments in the stored form for reinsertion because IMS/ESA Year 2000 Exit Tool converts it again during DL/I insertion.

**DBDSNAP control statement**

Specifies whether to generate a snap dump report of the DBD.

This control statement is optional. If you do not specify this control statement, DBDSNAP=NO is used.

<table>
<thead>
<tr>
<th>DBDSNAP=</th>
<th>NO</th>
<th>YES</th>
</tr>
</thead>
<tbody>
<tr>
<td>NO</td>
<td>Specifies that a DBD snap dump report is not generated.</td>
<td></td>
</tr>
<tr>
<td>YES</td>
<td>Specifies that a DBD snap dump report is generated.</td>
<td></td>
</tr>
</tbody>
</table>

**DBIOBUF control statement**

Specifies the number of tracks for the access method (VSAM or BSAM) buffers that are used for one database data set group when the database is read or written to.

IMS HP Load translates the specified number into the AMP= or DCB= (NCP=) parameter value internally. If the translated value exceeds the maximum allowable value for each access method, the translated value is overwritten. If you specify the AMP= or the DCB= (NCP=) parameter for the database DD statement, it precedes the DBIOBUF= specification.

This control statement is optional. If you do not specify this control statement, DBIOBUF=30 is used for VSAM, and DBIOBUF=15 is used for OSAM. In the case of HISAM, DBIOBUF=15 is used.
**DBIOBUF**

Specifies a left-aligned decimal number in the range of 1 - 999.

**Note:** IMS HP Load calculates the number of VSAM buffers that fit into 30 tracks according to the CI size, and specifies AMP=('BUFND=nnn, BUFFNI=100') in the KSDS database data set internally. BUFFNI=100 is used commonly for any CI size.

---

### DBRLBUF control statement

Specifies the number of database buffers (the number of tracks) to be used for one database data set group when the database is being read or written.

This statement is mutually exclusive with the BUFFERNO statement. Determine whether you use DBRLBUF to specify the amount of database buffers, or use BUFFERNO to specify the number of database buffers. With the DBRLBUF statement, you can specify the buffer amount more directly, without calculating the value using the CI size or the block size of each database data set.

Because the database buffers are acquired by the data set group, the total amount of database buffers is the sum of database buffers in each data set group.

This control statement is optional. If you do not specify this control statement, DBRLBUF=45 is assumed (that is, a buffer size of 45 tracks). In the case of HISAM, DBRLBUF=2 is used.

---

**Note:** You can reduce EXCPs by increasing the amount of database buffers. The amount of database buffers must be greater than the amount of access method buffers plus 10 tracks. If the amount of database buffers is insufficient, IMS HP Load internally adjusts the amount of database buffers. Select an appropriate value for the amount of database buffers based on your execution environment.

---

### EXTENT control statement

Specifies the maximum number of extensions of the OSAM data set that can be created. IMS HP Load checks the extension number of a reorganized data set after its reorganization. When the extension number is larger than what you specified on the EXTENT control statement, IMS HP Load issues a user abend.

This control statement is optional. If you do not specify this control statement, EXTENT=52 is used.

**Note:** If the OSAM data set is created on EAV, this control statement is ignored.
Specifies a left-aligned decimal number in the range of 1 - 60.

**FRSPC control statement**

Specifies whether IMS HP Load secures free space according to the specification of the FRSPC parameter in DBD.

This control statement is optional. If you do not specify this control statement, FRSPC=YES is used.

<table>
<thead>
<tr>
<th>FRSPC=</th>
<th>YES</th>
<th>NO</th>
</tr>
</thead>
<tbody>
<tr>
<td>YES</td>
<td>Free space is secured according to the specification of the DBD.</td>
<td></td>
</tr>
<tr>
<td>NO</td>
<td>Free space is not secured. The specification of the DBD is ignored.</td>
<td></td>
</tr>
</tbody>
</table>

**GDGBASE control statement**

Specifies whether to create a generation data group (GDG) base entry when a GDG base entry does not exist.

This control statement is optional. If you do not specify this control statement, GDGBASE=NO is used.

<table>
<thead>
<tr>
<th>GDGBASE=</th>
<th>YES</th>
<th>NO</th>
</tr>
</thead>
<tbody>
<tr>
<td>YES</td>
<td>If there is no GDG base entry for the generation data set that is to be created and if the generation data set is to be allocated dynamically, a GDG base entry is created.</td>
<td></td>
</tr>
<tr>
<td>NO</td>
<td>GDG base entry is not created.</td>
<td></td>
</tr>
</tbody>
</table>

Tips: The default value of the GDGBASE control statement can be changed. For details, see Chapter 17, “Setting installation default options,” on page 241.

**GDGLIMIT control statement**

Specifies the maximum number of generation data sets (GDS) that can be associated with the generation data group (GDG) that is to be created.

This specification is valid only when GDGBASE=YES is specified. This control statement is optional. If you do not specify this control statement, GDGLIMIT=10 is used.
GDGLIMIT=nnn

Specifies a left-aligned decimal number in the range of 1 - 255.

Tip: The default value of the GDGLIMIT control statement can be changed. For details, see Chapter 17, “Setting installation default options,” on page 241.

HPIO control statement

Specifies whether IMS HP Load uses the High Performance I/O Driver for output database data sets.

This control statement is optional. If you do not specify this control statement, the value for the HPIO option is determined as follows: if the libraries that are specified on the STEPLIB DD statement are not APF-authorized, HPIO=NO is used; otherwise, HPIO=YES is used.

HPIO=YES
The High Performance I/O Driver is used for output database data sets.

HPIO=NO
The High Performance I/O Driver is not used for output database data sets.

Notes:
• For the HISAM or SHISAM, HPIO=NO is assumed. If you specify HPIO=YES, it will be ignored.
• You must run this statement under program HPSRRC00, which is the IMS HP Load version of the region controller program.
• Under IMS region controller (PGM=DFSRRC00), HPIO=NO is assumed. If you specify HPIO=YES, it will be ignored.

Tip: The default value of the HPIO control statement can be changed. For details, see Chapter 17, “Setting installation default options,” on page 241.

ILDSBLD control statement

Specifies whether to build ILDS and how to build the ILDS.

This control statement is optional. If you do not specify this control statement, ILDSBLD=YES,DSPR is used.

Note: ILDSBLD=YES is obsolete. If you specify ILDSBLD=YES, ILDSBLD=YES,DSPR is used.
YES,DSPR
ILDS is rebuilt with using less amount of data space than the obsolete option ILDSBLD=YES.

In the process of rebuilding, the records of ILDS are arranged in sequence of the key value by using DFSORT. If you do not specify DD statements for HPSRILEA, HPSRILEB, and HPSRWKnn work data sets, which are used as SORTIN and SORTWK data sets, these data sets are allocated dynamically.

You can specify the data class and the storage class for HPSRILEA and HPSRILEB data sets by using ILEDATAC and ILESTORC control statements.

YES,INITONLY
The ILDS is initialized but no ILEs are built. Use ILDSBLD=YES,INITONLY only when the HALDB has no logical relationships and you want to rebuild PSINDEXes in a separate job by using IMS Index Builder.

C ILDS is rebuilt in the same manner as with IMS HD Reorganization Reload.

NO ILDS is not rebuilt. In this case, You must rebuild ILDS after the reorganization of database. To do this, scan the reorganized database by using the IMS HALDB Index/ILDS Rebuild Utility (DFSPREC0).

Tip: The default value of the ILDSBLD control statement can be changed. For details, see [Chapter 17, “Setting installation default options,” on page 241].

**ILEDATAC control statement**

Specifies the name of the data class for the HPSRILEA and HPSRILEB data sets.

This control statement is effective only when IMS HP Load reloads the records with the ILDSBLD=YES,DSPR option. If any other ILDSBLD option is used, ILEDATAC control statement is ignored.

This control statement is optional. This control statement does not have a default value.

<table>
<thead>
<tr>
<th>ILEDATAC=data_class_name</th>
</tr>
</thead>
</table>

*data_class_name*
Specifies the name of the data class for HPSRILEA and HPSRILEB data sets.

Tip: The default value of the ILEDATAC control statement can be changed. For details, see [Chapter 17, “Setting installation default options,” on page 241].
ILESTORC control statement

Specifies the name of the storage class for the HPSRILEA and HPARILEB data sets.

This control statement is effective only when IMS HP Load reloads the records with the ILDSBLD=YES,DSPR option. If any other ILDSBLD option is used, ILESTORC control statement is ignored.

This control statement is optional. This control statement does not have a default value.

```
ILESTORC=storage_class_name
```

*storage_class_name*

Specifies the name of the storage class for HPSRILEA and HPSRILEB data sets.

Tip: The default value of the ILESTORC control statement can be changed. For details, see Chapter 17, “Setting installation default options,” on page 241.

INPUT control statement

Specifies the input source of the HD unloaded data set and specifies whether IMS HP Load invokes a user exit routine.

This control statement is optional. If you do not specify this control statement, INPUT=DFSUINPT is used.

```
INPUT=DFSUINPT,exitname
```

IMS HP Load can reload an unloaded data set that is specified in the DFSUINPT DD statement in one of the following formats—created by IMS HP Unload—or in an equivalent format created by a compatible program:

- The HD format
- The HD format with compressed segments
- The short header format
- The communication industry standard format (see Restrictions)
- The communication industry partitioned format (see Restrictions)

Restrictions:

- If the database has a logical relationship with another database, IMS HP Load cannot reload a data set in either the communication industry standard format or the communication industry partitioned format.
• If the database is a HALDB, IMS HP Load cannot reload a data set in either the short header format or the communication industry standard format.
• If the database is a SHISAM DB, the communication industry standard format is not supported.
• If the database is a non-HALDB, IMS HP Load cannot reload a data set in the communication industry partitioned format.

IMS HP Load determines which format is being used; you do not have to specify it.

`exitname`
The name of the user exit routine. Specify this operand if you use the exit interface of IMS HP Load. The specified exit routine must reside in the program library (JOBLIB, STEPLIB, or LINKLIB).

`L`
Specify this operand only if the user exit routine is to be run under an environment of LE (Language Environment). This option is effective when the user exit routine is written in Enterprise COBOL for z/OS or Enterprise PL/I for z/OS. This option is not effective for Assembler.

**ITKBLOAD control statement**
Specifies the name of the load module library for IMS Tools Knowledge Base.

This control statement is used only when ITKBSRVR=server_name is specified. If you do not specify this control statement, ITKBLOAD=*NO will be used and IMS HP Load attempts to load the IMS Tools KB module from the STEPLIB libraries.

```
ITKBLOAD=NO
```

*NO IMS Tools KB modules are loaded from the private library or the system library of the job.

`library_name`
Specifies the name of the IMS Tools KB load module data set that is to be used by IMS HP Load.

**Requirement:** When STEPLIB needs to be APF-authorized, this library also needs to be APF-authorized.

**Tip:** The default value of the ITKBLOAD control statement can be changed. For details, see Chapter 17, “Setting installation default options,” on page 241.

**ITKBSRVR control statement**
Specifies the name of the IMS Tools KB server.

This control statement is optional. If you do not specify this control statement, ITKBSRVR=*NO is used.

**Requirement:** ITKBSRVR=server_name requires that an appropriate version of IMS Tools Knowledge Base is installed and that the server is started. For details, see
*NO  IMS HP Load does not store reports in the IMS Tool KB Output repository.

server_name  
IMS HP Load stores reports in the IMS Tools KB Output repository of the specified server.  

If ITKBSRVR=server_name is specified, IMS HP Load attempts to connect to the server and then to store the following reports in the Output repository of IMS Tools Knowledge Base:

- FRRIN Control Statements
- Segment Count Summary
- Segment Statistics
- Data Set Statistics
- Pointer Count Summary
- Pointer Statistics
- HALDB Process Summary

ERRC04  
IMS HP Load returns return code 4 when it encounters IMS Tools KB errors.

Tip: The default value of the ITKBSRVR control statement can be changed. For details, see Chapter 17, “Setting installation default options,” on page 241.

**LPART control statement**

Specifies the HALDB partitions to load.

Use the LPART control statement to specify the partitions to load in the following cases:

- When the database data sets are created by using the DDEFPDS DD statement (IDCAMS commands) and the input unloaded data set is not created by IMS HP Unload with the CHECKREC=YES option.
- When the database data sets are created by using the DDEFPDS DD statement (IDCAMS commands) and you have added new partitions after unloading the database.
- When you have added new partitions after unloading the database and you have one or more empty new partitions that you want to initialize with the PARTINIT=YES,LPART option.

Tip: To avoid errors, consider specifying this control statement only in these cases.
If the unload data sets contain segments for the partitions that are not specified on the LPAR control statement, IMS HP Load issues an HPSR3768W message and continues processing.

This control statement is optional and it does not have a default value. This control statement cannot be used for the Load API.

```
LPART=ALL
    partname
    (partname,nnnn
     partlist)
```

**ALL** Specifies to load all the partitions. Use this option only when you reorganize the entire database and when the input unloaded data set contains data for all the partitions.

**partname**
Specifies the name of the partition to load, or the first partition of several sequential partitions to load.

**nnnn** Specifies the number of sequential partitions to load. If this parameter is not specified, the default value of 1 is used.

For example, the following statements have the same meaning:

- LPART=PARTA
- LPART=(PARTA)
- LPART=(PARTA,1)

**Important:** The processing order of the partitions is determined by the HALDB partition selection exit routine or the HALDB high key values. The processing order does not depend on the partition names. When you process multiple partitions, be aware of the processing order.

**partlist**
Specifies lists of the partition names to load. The names must be separated by commas and enclosed in parentheses. You can specify up to 1001 names by using multiple statements as follows:

- LPART=(part1,part2,part3,part4,part5)
- LPART=(part6,part7,part8,part9,part10)

You can specify non-sequential partitions.

The processing order of the partitions is determined by the HALDB partition selection exit routine or the HALDB high key values. The specified order in the list does not affect the processing order.

**OADSPR control statement**
Specifies whether IMS HP Load reduces the usage of data space when processing the overflow area by using the High Performance I/O driver or by using an OVERFLOW DD. This specification is valid only for HDAM or PHDAM databases.
This control statement is optional. If you do not specify this control statement, OADSPR=NO is used.

YES    Reduces the usage of data space when processing the overflow area.
NO     Does not reduce the usage of data space when processing the overflow area.

Notes:

- If you specify both OADSPR=YES and HPIO=YES, IMS HP Load attempts to reduce the usage of data space by using the High Performance I/O driver. It can also reduce the time for reload except when IPR Type-A Image Copy processing is used for a (P)HDAM that has extremely large overflow area.
- If you specify both OADSPR=YES and HPIO=NO, IMS HP Load attempts to reduce the usage of data space by using an OVERFLOW DD as a temporary data set. If the DD statement is not specified in the JCL, IMS HP Load attempts to allocate it dynamically on a direct-access device. For details, see "JCL requirements" on page 35.
- This function might not be able to reduce the usage of data space depending on the database. Because IMS HP Load keeps the data in a data space until the information for pointers in the block are resolved.

Tip: The default value of the OADSPR control statement can be changed. For details, see Chapter 17, “Setting installation default options,” on page 241.

**PAD control statement**

Specifies the padding character to fill out the segments to adjust to the length defined in the DBD definition.

The segments are padded in the following cases:

- When a segment edit/compression exit is not used and the length of fixed-length segments is increased because the DBD definition changed.
- When a segment edit/compression exit is used and the length of fixed-length segments is shorter than the maximum number of bytes defined by the COMPRTN parameter in the DBD definition.
- When the length of variable-length segments is shorter than the minimum segment length defined in the DBD definition.

This control statement is optional. If you do not specify this control statement, PAD=X’00’ is used.
Specify a two digit hexadecimal code as the padding character. For example, PAD=X'FF'.

Tip: The default value of the PAD control statement can be changed. For details, see Chapter 17, “Setting installation default options,” on page 241.

**PARTINIT control statement**

Specifies whether to initialize HALDB partitions after reload. IMS HP Load initializes a HALDB partition when a HALDB partition does not contain any segment during reload. The PARTINIT control statement is valid only for a HALDB.

This control statement is optional. If you do not specify this control statement, PARTINIT=NO is used.

<table>
<thead>
<tr>
<th>NO</th>
<th>Specifies that any initialization of a HALDB partition is not performed after reload.</th>
</tr>
</thead>
<tbody>
<tr>
<td>YES</td>
<td>Specifies that initialization of a HALDB partition is performed for partitions that are not populated by any segment during reload.</td>
</tr>
<tr>
<td>PINITON</td>
<td>Specifies to initialize the HALDB partition when the flag in the DBRC RECON record for the partition shows that the partition needs to be initialized (PINIT). If you specify only PARTINIT=YES, PARTINIT=YES,PINITON is assumed.</td>
</tr>
<tr>
<td>UNLOAD</td>
<td>Specifies to initialize the HALDB partition when the partition information is included in the input unloaded data set. To use this option, the following conditions must be satisfied:</td>
</tr>
<tr>
<td></td>
<td>• The input unloaded data set has been created by IMS HPUnload and with the CHECKREC=YES option.</td>
</tr>
<tr>
<td></td>
<td>• The input unloaded data sets are not concatenated.</td>
</tr>
<tr>
<td>LPART</td>
<td>Specifies to initialize the HALDB partitions that are specified on the LPART control statement.</td>
</tr>
</tbody>
</table>

If you want to use IMS HP Load as a replacement for the HALDB partition data set initialization utility, specify the PARTINIT=YES control statement and DFSUINPT DD DUMMY in the IMS HP Load JCL, and run it.

IMS HP Load issues a warning message if the partition was not initialized by the following reasons:

• PARTINIT=YES,PINITON is specified, but the flag in the RECON shows that the partition does not need to be initialized (NOPINIT).
• PARTINIT=YES,UNLOAD is specified, but the partition information is not found in the unloaded data set.

The following conditions must be satisfied to issue the warning message:
• The input unloaded data set has been created by IMS HP Unload and with the CHECKREC=YES option.
• The input unloaded data sets are not concatenated.

Tip: The default value of the PARTINIT control statement can be changed. For details, see Chapter 17, “Setting installation default options,” on page 241.

PARTSTAT control statement
Specifies whether the partition-wide occurrence count report is generated. This specification is valid only for entire database reload of a partitioned database.

This control statement is optional. If you do not specify this control statement, PARTSTAT=YES is used.

YES Specifies that the partition-wide occurrence count report is generated.
NO Specifies that the partition-wide occurrence count report is not generated.

PROGMON control statement
Specifies whether IMS HP Load issues a progress monitoring message at the specified intervals.

This control statement is optional. If you do not specify this control statement, PROGMON=0 will be used (that is, no progress monitoring message is issued).

RAAFORMAT control statement
Specifies whether IMS HP Load is to format HDAM or PHDAM RAA blocks regardless of the number of blocks used. Once the overflow area is used, this parameter is ignored and the entire RAA is formatted. This specification is valid only for HDAM or PHDAM databases.
This control statement is optional. If you do not specify this control statement, RAAFORMAT=YES is used.

\[
\begin{array}{ll}
\text{RAAFORMAT} & \text{YES NO} \\
\text{RAAF} & \\
\end{array}
\]

The abbreviation RAAF= can be used for RAAFORMAT=.

**YES** The complete RAA is formatted.

**NO** The complete RAA is not formatted. The RAA is formatted through the last block used.

**RECNUM control statement**

Specifies the number of unloaded records to be reloaded. It helps to optimize the DFSORT main storage and the allocation of the intermediate storage data sets.

You can use this control statement only when you specify SORT=YES, SORT=COND, or SORTIDAM=YES.

This control statement is optional. This control statement does not have a default value.

\[
\begin{array}{l}
\text{RECNUM=nnnnnnnnnn} \\
\end{array}
\]

\text{nnnnnnnnnn}

Specifies a left-aligned decimal number in the range of 1 - 2147483647.

**REPAIRILK control statement**

Specifies whether to reassign indirect list keys (ILKs). In reassigning ILKs, IMS HP Load uses new RBA, partition ID, and reorganization number.

This control statement is optional. If you do not specify this control statement, REPAIRILK=NO is used.

\[
\begin{array}{ll}
\text{REPAIRILK} & \text{YES NO} \\
\end{array}
\]

**NO** ILKs are not reassigned.

**YES** ILKs, stored in the segment prefix, are reassigned by using new RBA, partition ID, and reorganization number.
Attention: If you want to reassign ILKs by specifying REPAIRILK=YES, ensure that the entire HALDB (that is, all the partitions of the HALDB) is reorganized with the REPAIRILK=YES option. Otherwise, you might encounter problems that might be caused by duplicate segment ILKs.

If you specify REPAIRILK=YES, you must rebuild the PSINDEX for the target HALDB because ILKs are changed.

Restriction: Logical relationships are not supported.

Tip: The default value of the REPAIRILK control statement can be changed. For details, see Chapter 17, “Setting installation default options,” on page 241.

RMBSEGSZ control statement
Specifies how IMS HP Load calculates the database record size to determine whether to put the records to the overflow area.

This control statement is optional. If you do not specify this control statement, RMBSEGSZ=NO is assumed.

YES Means the actual size of the variable length segment (the LL value) is used to calculate the database record size. This is applicable also for the compressed segment.

NO Means the MINBYTES value that is specified on the DBD is used to calculate the database record size.

This specification is valid for HDAM and PHDAM databases that have the variable length segment. If databases other than HDAM and PHDAM databases that have the variable length segments are specified, IMS HP Load ignores them.

SEARCH control statement
Specifies the limit value for a space search algorithm that finds a segment that does not fit in its home block. This specification is valid for HDAM or PHDAM databases.

IMS HP Load searches free space by either of the following methods:

• If the specified number is an unsigned value (nnn), IMS HP Load searches for free space the specified number (nnn) of adjacent blocks both forward and backward. IMS HP Load first searches backward, then forward.

• If the specified number is a negative value (-nnn), IMS HP Load searches for free space the specified number (nnn) of adjacent blocks only backward.

To search available free space for segment insertion, IMS HP Load simply ignores the SCAN parameter in DBD and uses the SEARCH parameter to process reloading faster.
This control statement is optional. If you do not specify this control statement, SEARCH=3 is used.

\[ \text{SEARCH}=\begin{array}{c} nn \\\text{-nnn} \\
\end{array} \]

\( nnn \) Specifies a left-aligned decimal number in the range of 0 - 999. IMS HP Load searches the specified number of adjacent blocks both forward and backward.

\( -nnn \) Specifies a left-aligned decimal number in the range of -0 - -999. IMS HP Load searches the specified number of adjacent blocks only backward.

0 and -0 have the same meaning, bypassing space search and putting the segment into the overflow area.

**SEQERROR control statement**

Specifies how to handle key sequence errors in the input unloaded data.

This control statement is optional. If you do not specify this control statement, SEQERROR=ABEND is used.

\[ \text{SEQERROR}=\begin{array}{c} \text{ABEND} \\
\text{SAVE},nnnn \\
\text{ACCEPT},nnnn \\
\end{array} \]

**ABEND**

Issues a user abend after a key sequence error.

**SAVE**

Continues processing after key sequence errors. Segments in error and their dependent segments are all placed in the SEQERROR data set.

If you specify this option, you must specify the SEQERROR DD statement.

You can write a DL/I program to insert segments saved in the SEQERROR data set into the database after database reload completes. A sample user application is provided in SHPSSAMP with the member name FABFSQEI.

If you use the SEQERROR=SAVE option for a database with a logical relationship, the SEQERROR data set might contain logical child segments in key sequence error. Do not reinsert such logical child segments with a user-written DL/I insert program; otherwise, you receive the DL/I status code II.

**Note:** When root keys of HDAM or PHDAM are not in the physical sequence order of the block and RAP numbers, IMS HP Load issues a user abend even if SEQERROR=SAVE is specified.
**nnnn** Specifies the maximum number of sequence errors. *nnnn* must be a left-aligned decimal number in the range of 1 - 32767. If the number of sequence errors exceeds this value, IMS HP Load issues a user abend.

If this value is not specified, IMS HP Load accepts any number of key sequence errors.

**ACCEPT**

Specifies that processing is to continue after key sequence errors of dependent segments and that the segments that contain errors are placed in the database with those errors. The root keys of HIDAM, HISAM, SHISAM, or PHIDAM should be in the proper ascending sequential order without duplication. The root keys of HDAM or PHDAM should be in the physical sequence order of the block and RAP numbers, and if a root segment that has duplicate key is found, it is placed in the synonym chain in the database. Details about each key sequence error are displayed in the Messages report.

**Note:** Even if you specify SEQERROR=ACCEPT, IMS HP Load issues a user abend under the following conditions:

- Root keys of HIDAM, HISAM, SHISAM, or PHIDAM are not in the proper ascending sequential order or have a duplicate key.
- Root keys of HDAM or PHDAM are not in the physical sequence order of the block and RAP numbers.

**nnnn** Specifies the maximum number of sequence errors. *nnnn* must be a left-aligned decimal number in the range of 1 - 32767. If the number of sequence errors exceeds this value, IMS HP Load issues a user abend.

If this value is not specified, IMS HP Load accepts any number of key sequence errors.

**Note:** If you specify SEQERROR=SAVE or ACCEPT, the numbers of occurrences of key sequence errors are listed in the Segment Count Summary report by segment type.

**SORT control statement**

Specifies whether to run the physical sequential sort processing before loading. This specification is valid only for HDAM and PHDAM databases.

The HDAM or PHDAM unloaded data must be in the order of hexadecimal value *bbbbbbr* that represents the result of randomizing module conversion. *bbbbb* is the block number and *rr* is the RAP number. If it is not in that order due to such reasons as changing randomizer, you must specify SORT=YES or SORT=COND.

This control statement is optional. If you do not specify this control statement, SORT=NO is used.
NO  Does not run the sort process.

YES  Runs physical sequential sorting before the reload process. It sorts in the physical sequence of the block, then the RAP numbers.

COND  Runs the physical sequential sorting depending on conditions. To use this option, the input unloaded data set must be created by IMS High Performance Unload with the CHECKREC=YES option and the format of the unloaded data set must be in the HD format or in the short header format.

The physical sequential sorting is done if any of the following is changed between the unload process and the reload process:

- The randomizing module name
- The number of root anchor points (RAP)
- The maximum BLK/CI number in RAA
- The starting position of the root key
- The length of the root key
- Whether the root segment has key sequence field
- Database or organization is changed from (P)HIDAM to (P)HDAM

The physical sequential sorting is done whenever the following conditions are satisfied:

- The input unloaded data set was not created by IMS HP Unload with the CHECKREC=YES option.
- The input unloaded data set was created in the communication industry standard format.
- The input unloaded data sets are concatenated.
- The Load API function is used.

**Requirement:** For the following changes, you must specify SORT=YES instead of SORT=COND, because these changes cannot be detected by IMS HP Load with SORT=COND:

- The logic of the randomizing module
- The name or logic of the HALDB partitions selection exit routine
- The partition selection string or the high key values

**Restrictions:**

- When processing HALDBs, this statement is used exclusively with the USERHDR and the USERHDRX control statements.
- The key value that is sorted by SORT=YES or SORT=COND cannot be changed by the user exit routine.
- The compressed root key cannot be sorted. Specify DECOMPRESS=YES in the IMS HP Unload utility to expand the compressed key.

**Note:** If a workspace of DFSORT needs to be adjusted, specify the SORTWKnm DD statements.
Tip: The default value of the SORT control statement can be changed. For details, see Chapter 17, “Setting installation default options,” on page 241.

**SORTHIDAM control statement**

Specifies whether to run the sort processing before loading a HIDAM or a PHIDAM.

The HIDAM or PHIDAM unloaded data must be in the order of root keys. If it is not in that order because the root keys are changed after unloading, you must specify SORTHIDAM=YES. This control statement is optional. If you do not specify this control statement, SORTHIDAM=NO is used.

<table>
<thead>
<tr>
<th>SORTHIDAM=</th>
<th>NO</th>
<th>YES</th>
</tr>
</thead>
</table>

- NO  Does not run the sort process.
- YES Runs the key sequential sort process before the reload process. The sort is done in the order of the root keys.

**Restrictions:**
- This control statement cannot be used for the Load API.
- The key value that is sorted by SORTHIDAM=YES cannot be changed by the user exit routine that is specified in the reload step.
- Compressed root keys cannot be sorted. Specify DECOMPRESS=YES for the IMS HP Unload job to expand the compressed keys.
- HPSRSVX2 exit cannot run with SORTHIDAM=YES.

**Note:** For a HIDAM database, DFSORT allocates intermediate storage data sets dynamically for the sort process with sizes estimated based on the primary allocation size of DBDS. For a PHIDAM database, the size is estimated based on the number of partitions. If you specify the number of unloaded records in the RECNUM control statement, this value supersedes other values and is used for estimation. If you need to adjust the size, specify the SORTWKnn DD statements explicitly.

**TWINSTAT control statement**

Specifies whether to print statistics about the twin chained segments.

The statistics about the twin chained segments are printed in the Twin Segment Statistics section in the Segment Statistics report. For an example of the report, see “Segment Statistics report” on page 93.

This specification is valid for HDAM, HIDAM, PHDAM, or PHIDAM.

This control statement is optional. If you do not specify this control statement, TWINSTAT=NO is used.
NO Statistics about the twin chained segments are not printed.
YES Statistics about the twin chained segments are printed.

If you process a HALDB with TWINSTAT=YES, PARTSTAT=YES is internally used. The statistics reports, which include the twin segment statistics section, are generated for each partition.

Tip: The default value of the TWINSTAT control statement can be changed. For details, see Chapter 17, “Setting installation default options,” on page 241.

**USEREXIT control statement**

Specifies whether IMS HP Load invokes a user exit routine.

This control statement is optional. If you do not specify this control statement, no user exit is invoked.

```
USEREXIT=exitname
```

*exitname*

The name of the user exit routine. The specified exit routine must reside in the program library (JOBLIB, STEPLIB, or LINKLIB).

You can use either the INPUT= or USEREXIT= statement to specify the name of the user exit routine. The result is the same.

If both of INPUT= and USEREXIT= are specified at the same time, the INPUT= statement is used. Unlike the INPUT control statement, L—for specifying the usage of Language Environment—cannot be specified with the USEREXIT control statement.

**USERHDR control statement**

Specifies the name of the element of the header portion of the unloaded record to read the customized unload file.

```
USERHDR=(element)
```

You can specify up to 20 elements.
**element**

Describes an element that constitutes the user header portion of the unloaded file. 17 kinds of elements are supported as listed in the following table. You can use any combination of them as parameters of the USERHDR control statement.

The order in which you specify the elements determines the order that the elements appear in the user header portion of the unloaded file.

If the specification exceeds one line, specify as follows:

```plaintext
//FRRIN DD *
USERHDR=(DBDNAME,SEGCODE,SEGLEV1,SEGNAME)
USERHDR=(SEGDATSZ,SEGKEY,FILLER11)
/*
```

The following table summarizes the elements that you can specify on the USERHDR control statement.

**Table 9. Description and values of the elements of USERHDR**

<table>
<thead>
<tr>
<th>Element name</th>
<th>Field written to unloaded file: length in bytes</th>
<th>Field written to unloaded file: format of contents</th>
<th>Field written to unloaded file: description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBDNAME</td>
<td>8</td>
<td>character</td>
<td>Contains the DBD name.</td>
</tr>
<tr>
<td>FILLERnn</td>
<td>1–99</td>
<td>integer</td>
<td>The nn suffix defines the length of the field. The field is filled with low-values (binary zeros).</td>
</tr>
<tr>
<td>HDRLEN</td>
<td>2</td>
<td>binary</td>
<td>If you use SEGKEY, SEGCKEY, or ROOTKEY, the field contains the length of the user header record. This field is required.</td>
</tr>
<tr>
<td>ROOTKEY</td>
<td>variable</td>
<td>varies</td>
<td>Contains the sequence field of the root segment.</td>
</tr>
<tr>
<td>ROOTRBA</td>
<td>4</td>
<td>binary</td>
<td>Contains the database RBA of the root segment.</td>
</tr>
<tr>
<td>SEGCKEY</td>
<td>variable</td>
<td>varies</td>
<td>Contains the fully concatenated key of the segment.</td>
</tr>
<tr>
<td>SEGCKYSZ</td>
<td>2</td>
<td>binary</td>
<td>Contains the length of the segment’s concatenated key.</td>
</tr>
<tr>
<td>SEGCODE</td>
<td>1</td>
<td>binary</td>
<td>Contains the segment code of the segment. Either this keyword or SEGNAME is required.</td>
</tr>
<tr>
<td>SEGDATSZ</td>
<td>2</td>
<td>binary</td>
<td>Contains the length of the segment’s data.</td>
</tr>
<tr>
<td>SEGDBYTE</td>
<td>1</td>
<td>binary</td>
<td>Contains the delete byte of the segment as found in the database.</td>
</tr>
<tr>
<td>SEGKEY</td>
<td>variable</td>
<td>varies</td>
<td>Contains the sequence field of the segment.</td>
</tr>
<tr>
<td>SEGKEYOF</td>
<td>2</td>
<td>binary</td>
<td>Contains the offset into the segment data where the segment’s key is located.</td>
</tr>
<tr>
<td>SEGKEYSZ</td>
<td>2</td>
<td>binary</td>
<td>Contains the length of the segment’s key.</td>
</tr>
<tr>
<td>SEGLEV1</td>
<td>1</td>
<td>binary</td>
<td>Contains the hierarchical level of the segment.</td>
</tr>
<tr>
<td>SEGLEV2</td>
<td>2</td>
<td>character</td>
<td>Contains the hierarchical level of the segment.</td>
</tr>
<tr>
<td>SEGNAME</td>
<td>8</td>
<td>character</td>
<td>Contains the name of the segment. Either this keyword or SEGCODE is required.</td>
</tr>
<tr>
<td>SEGRBA</td>
<td>4</td>
<td>binary</td>
<td>Contains the database RBA of the segment.</td>
</tr>
</tbody>
</table>
Important:
• You cannot duplicate keywords other than FILLERnn.
• You must specify the SEGNAME or the SEGCODE keyword. They must precede the SEGCKY, the SEGKEY, or the ROOTKEY element.
• If you use SEGKEY, SEGCKEY, or ROOTKEY, you must specify HDRLEN. It must precede the SEGKEY, the SEGCKEY, or the ROOTKEY element.

Restrictions: See “Restrictions for reloading user-defined header format records” on page 112 for restrictions that apply to using the USERHDR control statement.

USERHDRX control statement
Specifies whether IMS HP Load invokes the user header conversion exit routine.

This control statement is optional. If you do not specify this control statement, no user header conversion exit is invoked.

```
►►USERHDRX=exitname◄◄
```

**exitname**
The name of the user header conversion exit routine. The specified exit routine must reside in the program library (JOBLIB, STEPLIB, or LINKLIB).

IMS HP Load provides several exit routines, and you can specify one of them as the parameter of USERHDRX according to the format of the unloaded file, as shown in the following table.

<table>
<thead>
<tr>
<th>Exit routine name</th>
<th>Format of the unloaded file</th>
</tr>
</thead>
<tbody>
<tr>
<td>HPSRCVA3</td>
<td>*A3</td>
</tr>
<tr>
<td>HPSRCVA4</td>
<td>*A4</td>
</tr>
<tr>
<td>HPSRCVF1</td>
<td>*F1</td>
</tr>
<tr>
<td>HPSRCVF2</td>
<td>*F2</td>
</tr>
<tr>
<td>HPSRCVF3</td>
<td>*F3</td>
</tr>
<tr>
<td>HPSRCVF6</td>
<td>*F6</td>
</tr>
</tbody>
</table>

Restrictions: See “Restrictions for reloading user-defined header format records” on page 112 for restrictions that apply to using the USERHDRX control statement.

Related concepts:
Chapter 5, “Reloading user-defined header format records,” on page 111

WF1DSN control statement
Specifies the data set name for the DFSURWF1 data set when allocating the data set dynamically.

This control statement is optional and it does not have a default value. If you specify the DFSURWF1 DD statement, this control statement is ignored.
WF1DSN=

\[ data\_set\_name \]

Specifies the data set name for the DFSURWF1 data set. The name must be 1-44 alphanumeric or national character name ($, #, @).

\[ GDG\_base\_name.\&GDG \]

Specifies to create a DFSURWF1 data set as a generation data set. &GDG must be specified in the last position in the pattern.

\[ GDG\_base\_name \]

is used as the GDG base entry name. If the base entry does not exist and if you specify GDGBASE=YES, a new GDG base entry is created. The data set name generated must be a 1-44 alphanumeric or national character name ($, #, @).

**WF1REC control statement**

Specifies whether to write WF1 records in the DFSURWF1 or in HPSRSINDX output work data set, which is used as an input to resolve a logical or secondary index relationships. The WF1REC control statement is not valid for a HALDB.

This control statement is optional. If you do not specify this control statement, WF1REC=YES is used.

- **YES** If the database has a logical relationship or a secondary index, the records are written to the DFSURWF1 output work data set.
- **NO** WF1 records are not written to the DFSURWF1 or HPSRSINDX output work data set even if secondary index or logical relationships are defined in the database.

To resolve the secondary index, you have to run a database scan utility in a separate job step.

The DFSURWF1 data set is needed to run the IMS Database Prefix Resolution utility for logical relationships after this reload job step. For such non-HALDB databases, do not specify WF1REC=NO.

**Tip:** The default value of the WF1REC control statement can be changed. For details, see Chapter 17, “Setting installation default options,” on page 241.

**WF1SPACE control statement**

Specifies the space parameter for the DFSURWF1 data set when allocating the data set dynamically.
This specification is valid only when a WF1DSN control statement is specified. If you specify the DFSURWF1 DD statement, this control statement is ignored.

This control statement is optional and it does not have a default value. If you do not specify this control statement, IMS HP Load calculates the size based on the allocation size of the database data set.

WF1SPACE control statement
WF1SPACE=CYL,primary_alloc,secondary_alloc

CYL  Specifies that space is to be allocated in cylinders.
TRK  Specifies that space is to be allocated in tracks.

primary_alloc
   Specifies the primary space quantity. It must be a left-aligned decimal number in the range of 1 - 9999.

secondary_alloc
   Specifies the secondary space quantity. It must be a left-aligned decimal number in the range of 1 - 9999.

WF1UNIT control statement
WF1UNIT=SYSALLDA

SYSALLDA=unit_name

unit_name
   Specifies the device type or the device group name for the DFSURWF1 data set.

WF1VOL control statement
WF1VOL=unit_name

unit_name
   Specifies one or more volume names for the DFSURWF1 data set that is to be dynamically allocated.

This specification is valid only when a WF1DSN control statement is specified. If you specify the DFSURWF1 DD statement, this control statement is ignored.

This control statement is optional and does not have a default value.
**WF1VOL**

Specifies the volume names. You can make a nonspecific volume request by specifying an asterisk (*).

You can specify up to five volume names. Two or more volume names must be separated by commas and enclosed in parentheses. For example, WF1VOL=(VOLSR1,VOLSR2).

---

**WFPHLQ control statement**

Specifies the high-level qualifier for the WFP data sets.

This control statement is required only when you want to create WFP data sets. This control statement does not have a default value.

---

**WFPSPACE control statement**

Specifies the SPACE parameter for the WFP data sets when allocating WFP data sets dynamically.

This control statement is effective only when a WFPHLQ control statement is specified.

If you do not specify this control statement, IMS HP Load estimates the size based on the allocation size of the partition data set. This control statement is optional and it does not have a default value.

---

*volume_name*

Specifies the volume names. You can make a nonspecific volume request by specifying an asterisk (*).

You can specify up to five volume names. Two or more volume names must be separated by commas and enclosed in parentheses. For example, WF1VOL=(VOLSR1,VOLSR2).

---

**hlq**

Specifies the high-level qualifier for the WFP data sets. It must be 1-32 alphanumeric or national characters ($, #, @).

---

**CYL**

Specifies that space is to be allocated in cylinders.

**TRK**

Specifies that space is to be allocated in tracks.
**primary_alloc**
Specifies the primary space quantity. It must be a left-aligned decimal number in the range of 1 - 9999.

**secondary_alloc**
Specifies the secondary space quantity. It must be a left-aligned decimal number in the range of 1 - 9999.

**WFPUNIT control statement**
Specifies the device for the WFP data sets when allocating WFP data sets dynamically.

This control statement is effective only when a WFPHLQ control statement is specified.

This control statement is optional. If you do not specify this control statement, WFPUNIT=SYSALLDA is used.

```
WFPUNIT=unit_name
```

**unit_name**
Specifies the device type or the device group name for WFP data sets.

**WFPVOL control statement**
Specifies the volume names for the WFP data sets when allocating WFP data sets dynamically.

This control statement is effective only when a WFPHLQ control statement is specified.

This control statement is optional and it does not have a default value.

```
WFPVOL=(volume_name)*
```

**volume_name**
Specifies the volume names. You can indicate nonspecific volumes by designating an asterisk (*).

Two or more volume names must be separated by commas and enclosed in parentheses. For example, WFPVOL=(VOLSR1,VOLSR2). You can specify up to five volume names.
**WFPVOLCNT control statement**

Specifies the volume count for the WFP data sets when allocating WFP data sets dynamically.

This control statement is effective only when a WFPHLQ control statement is specified.

This control statement is optional and it does not have a default value.

```
WFPVOLCNT=nnn
```

### nnn

Specifies the number of allowable volumes for allocating the WFP data sets. The value is a decimal number in the range of 1 - 59 for a DASD data set.
Reports and statistics

IMS HP Load produces the reports and statistics in the SYSPRINT data set, the SNAP data set, and the HPSRAMSO data set.

SYSPRINT data set

The SYSPRINT data set contains the statistics reports.

The following reports are printed in this data set:
- FRRIN Control Statements
- Segment Count Summary
- Messages
- DBRC Messages
- Data Set Statistics
- Segment Statistics
- Pointer Count Summary
- Pointer Statistics
- HALDB Process Summary
- GDG Base Information
- WFP Process Summary

FRRIN Control Statements report

This report contains an echo back of the control statements you specified in the FRRIN data set. You can check the parameters used by IMS HP Load for this job step run.

The following figure shows an example of the FRRIN Control Statements report.
Figure 2. FRRIN Control Statements report (Part 1 of 2)
The control statements you specified in the FRRIN data set are listed below the scale in the report.

If a control statement turns out to be an error or needs a warning, an error message or a warning message is printed under the echo back of the control statement.

The "RUN TIME OPTIONS FOR THIS EXECUTION" part lists the options used for this utility run.

**DBD**  DBD name of the database.

**DBORG**

Database organization: HDAM, HIDAM, HISAM, SHISAM, PHDAM, PHIDAM; partitioned database or nonpartitioned database.

**OPTIONS FOR PROCESS CONTROL**

Options effective for process control.

**OPTIONS FOR DIAGNOSIS**

Options effective for diagnosis.

The date and time shown in the report heading tell when the reorganization reload was completed.
Segment Count Summary report

This report provides the balance between input and output segment occurrences of each segment type.

The following figure shows an example of the Segment Count Summary report for a PHDAM database.

All values in the report are decimal, and each line describes one segment type in the database.

---

IMS HIGH PERFORMANCE LOAD
5655-M26

"SEGMENT COUNT SUMMARY"

DBNAME = PHD00300  PARTITIONS = 5  ORG = PHDAM  ACCESS METHOD = OSAM

*** SEGMENT COUNT ACROSS ALL PARTITIONS PROCESSED ***

<table>
<thead>
<tr>
<th>SEGNAME</th>
<th>LVL</th>
<th>DSG</th>
<th>PARENT</th>
<th>READ</th>
<th>RELOADED</th>
<th>DELETED BY RANDOMIZER</th>
<th>DELETED BY USER EXIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOTLEV1</td>
<td>01</td>
<td>01</td>
<td>ROOTLEV1</td>
<td>40</td>
<td>40</td>
<td>0</td>
<td>N/A</td>
</tr>
<tr>
<td>DEP1LEV2</td>
<td>02</td>
<td>01</td>
<td>ROOTLEV1</td>
<td>40</td>
<td>40</td>
<td>0</td>
<td>N/A</td>
</tr>
<tr>
<td>DEP2LEV2</td>
<td>02</td>
<td>01</td>
<td>ROOTLEV1</td>
<td>40</td>
<td>40</td>
<td>0</td>
<td>N/A</td>
</tr>
<tr>
<td>DEP3LEV3</td>
<td>03</td>
<td>02</td>
<td>DEP2LEV2</td>
<td>80</td>
<td>80</td>
<td>0</td>
<td>N/A</td>
</tr>
<tr>
<td>DEP4LEV4</td>
<td>04</td>
<td>02</td>
<td>DEP3LEV3</td>
<td>160</td>
<td>160</td>
<td>0</td>
<td>N/A</td>
</tr>
<tr>
<td>DEP5LEV2</td>
<td>02</td>
<td>01</td>
<td>ROOTLEV1</td>
<td>40</td>
<td>40</td>
<td>0</td>
<td>N/A</td>
</tr>
<tr>
<td>DEP6LEV3</td>
<td>03</td>
<td>02</td>
<td>DEP5LEV2</td>
<td>80</td>
<td>80</td>
<td>0</td>
<td>N/A</td>
</tr>
<tr>
<td>DEP7LEV4</td>
<td>04</td>
<td>02</td>
<td>DEP6LEV3</td>
<td>160</td>
<td>160</td>
<td>0</td>
<td>N/A</td>
</tr>
<tr>
<td>DEP8LEV4</td>
<td>04</td>
<td>03</td>
<td>DEP6LEV3</td>
<td>160</td>
<td>160</td>
<td>0</td>
<td>N/A</td>
</tr>
<tr>
<td>DEP9LEV5</td>
<td>05</td>
<td>03</td>
<td>DEP8LEV4</td>
<td>320</td>
<td>320</td>
<td>0</td>
<td>N/A</td>
</tr>
<tr>
<td>TOTAL</td>
<td></td>
<td></td>
<td></td>
<td>1,120</td>
<td>1,120</td>
<td>0</td>
<td>N/A</td>
</tr>
</tbody>
</table>

ILDS RECORDS = 21

*** SEGMENT COUNT IN HDAM OVERFLOW AREA ***

<table>
<thead>
<tr>
<th>SEGNAME</th>
<th>OCCURRENCES</th>
<th>PERCENTAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOTLEV1</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>DEP1LEV2</td>
<td>40</td>
<td>100.0</td>
</tr>
<tr>
<td>DEP2LEV2</td>
<td>40</td>
<td>100.0</td>
</tr>
<tr>
<td>DEP3LEV3</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>DEP4LEV4</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>DEP5LEV2</td>
<td>40</td>
<td>100.0</td>
</tr>
<tr>
<td>DEP6LEV3</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>DEP7LEV4</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>DEP8LEV4</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>DEP9LEV5</td>
<td>0</td>
<td>0.0</td>
</tr>
<tr>
<td>TOTAL</td>
<td>120</td>
<td></td>
</tr>
</tbody>
</table>

---

Figure 4. Segment Count Summary report (for a PHDAM database)

This report consists of two parts: the segment count in the entire database part, and the segment count in the HDAM, PHDAM, or HISAM overflow area part.
The following list describes each item in the Segment Count Summary report. The items are common to HDAM, HIDAM, HISAM, and SHISAM databases, except that the overflow area statistics items are available only for HDAM, PHDAM, and HISAM databases.

<table>
<thead>
<tr>
<th>Item Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SEGNAME</strong></td>
<td>The name of the segment.</td>
</tr>
<tr>
<td><strong>LVL</strong></td>
<td>The hierarchical level of the segment.</td>
</tr>
<tr>
<td><strong>DSG</strong></td>
<td>The data set group number.</td>
</tr>
<tr>
<td><strong>PARENT</strong></td>
<td>The name of the parent segment.</td>
</tr>
<tr>
<td><strong>READ</strong></td>
<td>The number of segment occurrences that are read by IMS HP Load for this database.</td>
</tr>
<tr>
<td><strong>RELOADED</strong></td>
<td>The number of segment occurrences that IMS HP Load reloads for this database.</td>
</tr>
<tr>
<td><strong>DELETED BY RANDOMIZER</strong></td>
<td>The number of segment occurrences that are deleted because of a return code of 4 from the HDAM randomizer.</td>
</tr>
<tr>
<td><strong>DELETED BY USER EXIT</strong></td>
<td>The number of deleted segment occurrences due to a return code 8 or 12 from the user exit routine.</td>
</tr>
<tr>
<td><strong>ILDS RECORDS</strong></td>
<td>The number of ILE records in the ILDS for this partition.</td>
</tr>
</tbody>
</table>

The segment count in the HDAM overflow area part contains the following items:

<table>
<thead>
<tr>
<th>Item Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OCCURRENCES</strong></td>
<td>The number of occurrences of the segment.</td>
</tr>
<tr>
<td><strong>PERCENTAGE</strong></td>
<td>The percentage of the occurrences in the overflow versus the entire occurrence of this segment type.</td>
</tr>
</tbody>
</table>

The following figure shows an example of the report for a PHIDAM database. The segment count in PHIDAM overflow area part is not included in this report.
The following figure shows an example of the report when the SEQERROR control statement is specified for bypassing key sequence errors.

Figure 5. Segment Count Summary report (for a PHIDAM database)

The following figure shows an example of the report when the SEQERROR control statement is specified for bypassing key sequence errors.

Figure 6. Segment Count Summary report (when SEQERROR is specified)
SEGNAME
The name of the segment.

KEY SEQUENCE ERROR OCCURRENCES
The number of occurrences of key sequence errors for this segment type.

KEY SEQUENCE ERROR PERCENTAGE
The percentage of the occurrences of key sequence errors within the entire occurrences of this segment type.

OUTPUT IN SEQERROR DS OCCURRENCES
The number of segment occurrences saved in the SEQERROR data set for this segment type.

OUTPUT IN SEQERROR DS PERCENTAGE
The percentage of segment occurrences saved in the SEQERROR data set within the entire occurrences of this segment type.

Messages report
The messages report contains the messages about key sequence.

The following figures show examples of the Messages report that contain messages HPSR0028I, HPSR0034I, and HPSR3634E.
HPSR0034I A SEQUENCE FIELD IS NOT UNIQUE SEGMENT=ROOTLEV1 FIELD=ROOTNO
HPSR0028I 2 DB RECORDS PROCESSED
HPSR0034I A SEQUENCE FIELD IS NOT UNIQUE SEGMENT=ROOTLEV1 FIELD=ROOTNO
HPSR0028I 3 DB RECORDS PROCESSED
HPSR0034I A SEQUENCE FIELD IS NOT UNIQUE SEGMENT=ROOTLEV1 FIELD=ROOTNO
HPSR0028I 4 DB RECORDS PROCESSED
HPSR0034I A SEQUENCE FIELD IS NOT UNIQUE SEGMENT=ROOTLEV1 FIELD=ROOTNO
HPSR0028I 5 DB RECORDS PROCESSED
HPSR0034I A SEQUENCE FIELD IS NOT UNIQUE SEGMENT=ROOTLEV1 FIELD=ROOTNO
HPSR0028I 6 DB RECORDS PROCESSED
HPSR0034I A SEQUENCE FIELD IS NOT UNIQUE SEGMENT=ROOTLEV1 FIELD=ROOTNO
HPSR0028I 7 DB RECORDS PROCESSED
HPSR0034I A SEQUENCE FIELD IS NOT UNIQUE SEGMENT=ROOTLEV1 FIELD=ROOTNO
HPSR0028I 8 DB RECORDS PROCESSED
HPSR0034I A SEQUENCE FIELD IS NOT UNIQUE SEGMENT=ROOTLEV1 FIELD=ROOTNO
HPSR0028I 9 DB RECORDS PROCESSED
HPSR0034I A SEQUENCE FIELD IS NOT UNIQUE SEGMENT=ROOTLEV1 FIELD=ROOTNO
HPSR0028I 10 DB RECORDS PROCESSED
HPSR0034I A SEQUENCE FIELD IS NOT UNIQUE SEGMENT=ROOTLEV1 FIELD=ROOTNO
HPSR0028I 11 DB RECORDS PROCESSED
HPSR0034I A SEQUENCE FIELD IS NOT UNIQUE SEGMENT=ROOTLEV1 FIELD=ROOTNO
HPSR0028I 12 DB RECORDS PROCESSED
HPSR0034I A SEQUENCE FIELD IS NOT UNIQUE SEGMENT=ROOTLEV1 FIELD=ROOTNO
HPSR0028I 13 DB RECORDS PROCESSED
HPSR0034I A SEQUENCE FIELD IS NOT UNIQUE SEGMENT=ROOTLEV1 FIELD=ROOTNO
HPSR0028I 14 DB RECORDS PROCESSED
HPSR0034I A SEQUENCE FIELD IS NOT UNIQUE SEGMENT=ROOTLEV1 FIELD=ROOTNO
HPSR0028I 15 DB RECORDS PROCESSED
HPSR0034I A SEQUENCE FIELD IS NOT UNIQUE SEGMENT=ROOTLEV1 FIELD=ROOTNO
HPSR0028I 16 DB RECORDS PROCESSED
HPSR0034I A SEQUENCE FIELD IS NOT UNIQUE SEGMENT=ROOTLEV1 FIELD=ROOTNO
HPSR0028I 17 DB RECORDS PROCESSED
HPSR0034I A SEQUENCE FIELD IS NOT UNIQUE SEGMENT=ROOTLEV1 FIELD=ROOTNO
HPSR0028I 18 DB RECORDS PROCESSED
HPSR0034I A SEQUENCE FIELD IS NOT UNIQUE SEGMENT=ROOTLEV1 FIELD=ROOTNO
HPSR0028I 19 DB RECORDS PROCESSED
HPSR0034I A SEQUENCE FIELD IS NOT UNIQUE SEGMENT=ROOTLEV1 FIELD=ROOTNO
HPSR0028I 20 DB RECORDS PROCESSED

Figure 7. Messages report (HPSR0028I and HPSR0034I)

HPSR3634E A ROOT SEGMENT OF HIDAM/PHIDAM IS OUT OF SEQUENCE
DBD=HIVSDB10 SEGMENT=ROOTLEV1 FIELD=ROOTKEY1

000000000001
FFFFFFFFFFFFFF
000000000001

Figure 8. Messages report (HPSR3634E)
DBRC Messages report
This report contains the DBRC messages that are issued when DBRC=Y is specified for non-HALDB.

The following figure shows an example of the DBRC Messages report.

Data Set Statistics report
This report provides physical statistics about the database by each data set group.

The following figure shows an example of the Data Set Statistics report for an HDAM database.
"DATA SET STATISTICS"

DBNAME = DBHD0030  DS GROUPS = 3  ORG = HDAM  ACCESS METHOD = ESDS
RMNAME = DFSHD40  MAX INSERT = 40  NBR OF RAPS = 5  MAX RBN = 500

DS GROUP = 001 OF 003  DDNAME = DDHD0031  BLKSIZE = 4,096  LRECL = 4,089
DISTRIBUTED FREE SPACE = (000,00)  SCAN = 3
THRESHOLD OF LIMITED FREE SPACE BLOCK = 2,000 (LARGE SEGSZE 2,000)
NUMBER OF TOTAL BLOCKS = 501
NUMBER OF BITMAP BLOCKS = 1
NUMBER OF LIMITED FREE SPACE BLOCKS = 1
NUMBER OF FULL BLOCKS = 1
NUMBER OF FREE BLOCKS = 489
AVERAGE SIZE OF FREE SPACE ELEMENT = 4062.6

NUMBER OF HDAM ROOT SYNONYM CHAINS = 0
AVERAGE NUMBER OF ROOTS / SYNONYM CHAIN = 0.0
NUMBER OF ROOTS NOT ON SYNONYM CHAINS = 10
NUMBER OF ROOTS ON SYNONYM CHAINS = 0
NUMBER OF ROOTS OUT OF RANDOMIZED BLOCK = 0
NUMBER OF HDAM RAPS USED = 10
NUMBER OF HDAM RAPS UNUSED = 2,490
NUMBER OF HDAM OVERFLOW BLOCKS = 1

NUMBER OF SEQUENTIAL WRITES = 502
NUMBER OF DIRECT READS = 1
NUMBER OF DIRECT WRITES = 1

NUMBER OF DSPB ROUTINGS = 0
HIGHEST LENGTH OF DATA IN DSPB = 0
AVERAGE LENGTH OF DATA IN DSPB = 0.0

DS GROUP = 002 OF 003  DDNAME = DDHD0032  BLKSIZE = 512  LRECL = 505
DISTRIBUTED FREE SPACE = (000,00)  SCAN = 3
NUMBER OF TOTAL BLOCKS = 3
NUMBER OF BITMAP BLOCKS = 1
NUMBER OF LIMITED FREE SPACE BLOCKS = 2
NUMBER OF FULL BLOCKS = 1
NUMBER OF FREE BLOCKS = 0
AVERAGE SIZE OF FREE SPACE ELEMENT = 133.0

NUMBER OF SEQUENTIAL WRITES = 4
NUMBER OF DIRECT READS = 1
NUMBER OF DIRECT WRITES = 1

NUMBER OF DSPB ROUTINGS = 0
HIGHEST LENGTH OF DATA IN DSPB = 0
AVERAGE LENGTH OF DATA IN DSPB = 0.0

DS GROUP = 003 OF 003  DDNAME = DDHD0033  BLKSIZE = 1,024  LRECL = 1,017
DISTRIBUTED FREE SPACE = (000,00)  SCAN = 3

Figure 10. Data Set Statistics report (for an HDAM database) (Part 1 of 2)
"DATA SET STATISTICS"

<table>
<thead>
<tr>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUMBER OF TOTAL BLOCKS</td>
<td>5</td>
</tr>
<tr>
<td>NUMBER OF BITMAP BLOCKS</td>
<td>1</td>
</tr>
<tr>
<td>NUMBER OF LIMITED FREE SPACE BLOCKS</td>
<td>4</td>
</tr>
<tr>
<td>NUMBER OF FULL BLOCKS</td>
<td>1</td>
</tr>
<tr>
<td>NUMBER OF FREE BLOCKS</td>
<td>0</td>
</tr>
<tr>
<td>AVERAGE SIZE OF FREE SPACE ELEMENT</td>
<td>161.5</td>
</tr>
<tr>
<td>NUMBER OF SEQUENTIAL WRITES</td>
<td>6</td>
</tr>
<tr>
<td>NUMBER OF DIRECT READS</td>
<td>1</td>
</tr>
<tr>
<td>NUMBER OF DIRECT WRITES</td>
<td>1</td>
</tr>
<tr>
<td>NUMBER OF DSPB ROUTINGS</td>
<td>0</td>
</tr>
<tr>
<td>HIGHEST LENGTH OF DATA IN DSPB</td>
<td>0</td>
</tr>
<tr>
<td>AVERAGE LENGTH OF DATA IN DSPB</td>
<td>0.0</td>
</tr>
</tbody>
</table>

Figure 11. Data Set Statistics report (for an HDAM database) (Part 2 of 2)

The report shows the following items for each data set group. These items are common to HDAM, HIDAM, HISAM, SHISAM, PHDAM, and PHIDAM databases, except for those specified.

- The first line contains the database information found in the DBD, such as the database name, the number of data set groups, the organization, and the access method. If the organization of the reloaded database is HISAM or SHISAM, the ddname, block size, and record length are reported in these lines.
- The second line provides information about the randomizing module. This information is provided only for an HDAM or PHDAM database.
  - RMNAME specifies the 1- to 8-character alphanumeric name of a randomizing module.
  - MAX INSERT is the maximum number of bytes of a database record that can be inserted into the root addressable area.
  - NBR OF RAPS specifies the number of root anchor points desired in each control interval or block in the root addressable area.
  - MAX RBN is the maximum relative block number value.
- The first line of each data set group indicates the data set information, such as data set group number, ddname, block size, and record length.

**DISTRIBUTED FREE SPACE**

The free block frequency factor (fbff) and the free space percentage factor (fspf). fbff can be any integer value in the range of 0 - 100 except fbff=1. The fspf specifies the minimum percentage of each control interval or block that is to be left as free space. fspf is in the range of 0 - 99.

**THRESHOLD OF LIMITED FREE SPACE BLOCK**

The threshold value used to set bitmap that indicates whether each block or CI has a free space element (FSE) that is large enough. If BMSETDSG control statement is specified, the specified value is shown instead of the size of the largest segment in the data set group.

**SCAN**

The number of direct-access device cylinders to be scanned when searching for available storage space during segment insertion operations.
NUMBER OF TOTAL BLOCKS  
The total number of blocks or CIs written in the data set. If this data set is a VSAM data set, it does not include record 0.

NUMBER OF BITMAP BLOCKS  
The total number of bitmap blocks or CIs written in the data set.

NUMBER OF LIMITED FREE SPACE BLOCKS  
The number of blocks or CIs that do not have a free space element (FSE) large enough to hold the largest segment in the data set group.

NUMBER OF FULL BLOCKS  
The number of blocks or CIs that do not have enough free space for an FSE. An FSE requires a minimum of 8 bytes.

NUMBER OF FREE BLOCKS  
The number of blocks or CIs that do not have any segments in the data set group.

AVERAGE SIZE OF FREE SPACE ELEMENT  
The average amount of free space in those blocks that are not completely full. This average includes those blocks that are completely empty.

NUMBER OF SEQUENTIAL WRITES  
The number of sequential write operations to the database data set.

NUMBER OF DIRECT READS  
The number of direct read operations to the database data set.

NUMBER OF DIRECT WRITES  
The number of direct write operations to the database data set.

The following items are printed only for HDAM databases:

NUMBER OF HDAM ROOT SYNONYM CHAINS  
The number of HDAM root anchor points (RAPs) that have more than one root segment chained off it.

AVERAGE NUMBER OF ROOTS / SYNONYM CHAINS  
The average number of HDAM root segments per synonym chain.

NUMBER OF ROOTS NOT ON SYNONYM CHAINS  
The number of HDAM root segments that are not on any synonym chain.

NUMBER OF ROOTS ON SYNONYM CHAINS  
The number of HDAM root segments that are on one of synonym chains. A relatively high number here might have an impact on performance.

NUMBER OF ROOTS OUT OF RANDOMIZED BLOCK  
The number of root segments that were randomized to given blocks, but there was not enough space in the blocks to insert the segments. Those root segments that cannot be inserted in the root addressable area are placed in the overflow area. A relatively high number here might have an impact on performance.

NUMBER OF HDAM RAPS USED/UNUSED  
The number of total RAPs that are used or not used. The number of RAPs used versus the number of root segments is one indication of the efficiency of the randomizing routine. Comparison of the number of root segments with the number of RAPs used and unused gives you one indication of how well your HDAM randomizer is working. It does not indicate how uniformly the roots are distributed.
NUMBER OF HDAM OVERFLOW BLOCKS
The number of blocks or CIs in the overflow area.

The following items are printed only for HISAM databases that have more than one hierarchical segment level.

NUMBER OF RECORDS WITH OVERFLOW CHAINS
The number of roots with overflow chains.

MAX NUMBER OF OVERFLOW CHAINS
The maximum number of VSAM ESDS roots chained off from one VSAM KSDS root.

MIN NUMBER OF OVERFLOW CHAINS
The minimum nonzero number of VSAM ESDS roots chained off from one VSAM KSDS root.

AVERAGE NUMBER OF OVERFLOW CHAINS
The average number of VSAM ESDS roots chained off from one VSAM KSDS root.

NUMBER OF RECORDS WITH NO OVERFLOW CHAINS
The number of roots without dependent chains.

LONGEST DB RECORD LENGTH
The largest database record (in bytes) that has no overflow chains.

SHORTEST DB RECORD LENGTH
The shortest database record (in bytes) that has no overflow chains.

AVERAGE DB RECORD LENGTH
Average database record length (in bytes) that has no overflow chains.

NUMBER OF DSPB ROUTINGS
The number of times that data is written to and from the data space buffer (DSPB).

HIGHEST LENGTH OF DATA IN DSPB
The maximum length of the data in DSPB.

AVERAGE LENGTH OF DATA IN DSPB
The average length of the data written into DSPB.

Segment Statistics report
This report provides various information about all the segments in the database.

The following figure shows an example of the Segment Statistics report for an HDAM database.

All values in the report are decimal and each line describes one segment type in the database.

This report consists of two parts:
• SEGMENT STATISTICS part contains statistics on all segments
• VARIABLE LENGTH SEGMENT STATISTICS part contains variable-length segment statistics
The following figure shows another example of the Segment Statistics report. This example is for a HALDB partition.

The report contains the SEGMENT STATISTICS part and the VARIABLE LENGTH SEGMENT STATISTICS part. When TWINSTAT=YES is specified, the report also contains the TWIN SEGMENT STATISTICS part, which shows statistics about twin chained segments.
The following list describes each item in the Segment Statistics report. The items are common to both HDAM and HIDAM databases.

**SEGNAME**
The name of the segment.

**LVL**
The hierarchical level of the segment.

**DSG**
The data set group number.

**LENGTH OF PRFX**
The length of the segment prefix.

**LENGTH OF DATA**
The length of the segment data. The letter V following the number indicates that the segment has a variable length and that this data has the average data length of those occurrences found in the database. An asterisk

---

### VARIABLE LENGTH SEGMENT STATISTICS

<table>
<thead>
<tr>
<th>SEGNAME</th>
<th>LENGTH OF OCCUR (VERSUS MIN)</th>
<th>COMPRESSION ROUTINE FACTOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEP31LV3</td>
<td>4 40</td>
<td>DFSCMPX0 + 13.4</td>
</tr>
<tr>
<td>DEP41LV2</td>
<td>50 100</td>
<td>DFSCMPX0 + 42.7</td>
</tr>
<tr>
<td>DEP51LV3</td>
<td>100 200</td>
<td>N/A</td>
</tr>
<tr>
<td>DEP61LV3</td>
<td>70 140</td>
<td>N/A</td>
</tr>
</tbody>
</table>

---

### TWIN SEGMENT STATISTICS

<table>
<thead>
<tr>
<th>SEGNAME</th>
<th>NUMBER OF CHAINS</th>
<th>MAX CHAIN LENGTH</th>
<th>AVG CHAIN LENGTH</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOT1LV1</td>
<td>N/A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEP11LV2</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DEP21LV2</td>
<td>1</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>DEP31LV3</td>
<td>4</td>
<td>20</td>
<td>7.8</td>
</tr>
<tr>
<td>DEP41LV2</td>
<td>2</td>
<td>6</td>
<td>4.0</td>
</tr>
<tr>
<td>DEP51LV3</td>
<td>3</td>
<td>3</td>
<td>2.3</td>
</tr>
<tr>
<td>DEP61LV3</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
</tr>
</tbody>
</table>
(*) following the number indicates that the segment is padded with 1 byte to make it an even length in the database. IMS does not write odd-length segments in HD-type databases.

**OCCURRENCES**
The number of segment occurrences in the database. The number on the row TOTAL indicates the total number of segments in the database.

**NONFREE SPACE (%)**
This field indicates what percentage of the total number of bytes occupied by the database segments (prefix and data) is occupied by the segments of this type. This percentage should total 100%, but it might not because of rounding.

**AVERAGE RCD SIZE**
This field indicates the contribution to the average database record size by this segment type. This value is the product of the segment length (length of prefix and data) by the average occurrences per root segment. The total number in this column shows the average length of the whole database record. Since the value of this column is rounded off, the total value of each line might not be equal to the number shown by TOTAL.

\[
\text{AVERAGE RCD SIZE} = \left(\text{LENGTH OF PRFX} + \text{LENGTH OF DATA}\right) \times \text{AVERAGE OCCURRENCES PER ROOT}
\]

**AVERAGE OCCURRENCES PER PRNT**
The average number of occurrences of this segment type per immediate parent. It is also the average number of twin chain occurrences.

**AVERAGE OCCURRENCES PER ROOT**
The average number of occurrences of this segment type per root segment (LVL=01). It is also the average number of occurrences in a database record.

The variable-length segment statistics part provides further information about each variable-length segment with the following items:

**SEGNAME**
The name of the variable-length segment.

**LENGTH OF MIN**
The minimum data length of the segment occurrences.

**LENGTH OF MAX**
The maximum data length of the segment occurrences.

**OCCURRENCES (VERSUS MIN) LESS**
The number of occurrences of the segments that have a length less than the minimum data length of the segment. These segments might not have the data portion separated from the segments.

**OCCURRENCES (VERSUS MIN) EQUAL**
The number of occurrences of the segments that have a length equal to the minimum data length of the segment.

**OCCURRENCES (VERSUS MIN) GREATER**
The number of occurrences of the segments that have a length greater than the minimum data length of the segment.

**COMPRESSION ROUTINE**
The name of segment edit/compression routine, if specified for the segment. This field is blank if no segment edit/compression routine is specified for the segment.
**COMPRESSION FACTOR**

The compression factor shown as a percentage. The compression factor is calculated by the following formula:

\[
\text{Compression factor} = \frac{\text{Bytes after expansion} - \text{Bytes before expansion}}{\text{Bytes after expansion}}
\]

where “Bytes after expansion” is the sum of segment lengths after expansion. “Bytes before expansion” is the sum of segment lengths before expansion.

"0.0" is shown if the segment has no occurrence.

"N/A" is shown if no segment edit/compression routine is specified for the segment.

The twin chain statistics part provides further information about twin chained segments with the following items:

- **SEGNAME**
  The name of the segment.

- **NUMBER OF CHAINS**
  The total number of twin chain occurrences for the segment type.
  N/A is shown if a twin pointer is not defined for the segment.

- **MAX CHAIN LENGTH**
  The maximum length of the twin chains.

- **AVG CHAIN LENGTH**
  The average length of the twin chains.

**Pointer Count Summary report**

This report contains a list of all the pointers to a user-specified target, and various totals for each segment pointer type.

The following figure shows an example of the Pointer Count Summary report for an HDAM database.
The following list describes the fields in the Pointer Count Summary report. All values in the report are decimal.

**SOURCE SEGMENT**
The segment name that has one or more pointers in its prefix.

**TARGET SEGMENT**
The segment name that is pointed by the source segment.

**PTR TYPE**
The type of pointer used. For each pointer in a prefix, a line of information is printed. If the pointer type is HF or HB, the TARGET SEGMENT field contains an asterisk (*).

**NBR OF SOURCE SEGMENTS ZERO PTR**
The number of occurrences of the source segment for which this pointer is zero.

**NBR OF SOURCE SEGMENTS WITH PTR**
The number of occurrences of the source segment for which this pointer has an RBA. If the pointer type is PCF, this number indicates the number of physical twin chains in the database for the target segment type.

**Note:** If hierarchical pointers are used, the target segment name is not printed because hierarchical pointers can point to different segment types.

<table>
<thead>
<tr>
<th>SOURCE SEGMENT</th>
<th>TARGET SEGMENT</th>
<th>PTR TYPE</th>
<th>NBR OF SOURCE SEGMENTS ZERO PTR</th>
<th>NBR OF SOURCE SEGMENTS WITH PTR</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOTLEV1</td>
<td>ROOTLEV1</td>
<td>PTF</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>ROOTLEV1</td>
<td>PTB</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>DEP1LEV2</td>
<td>PCF</td>
<td>2</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>DEP2LEV2</td>
<td>PCF</td>
<td>2</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>DEP4LEV2</td>
<td>PCF</td>
<td>2</td>
<td>8</td>
</tr>
<tr>
<td>DEP1LEV2</td>
<td>DEP1LEV2</td>
<td>PTF</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>DEP1LEV2</td>
<td>PTB</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>DEP2LEV2</td>
<td>DEP2LEV2</td>
<td>PTF</td>
<td>8</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>DEP3LEV3</td>
<td>PCF</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>DEP3LEV3</td>
<td>PCL</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>DEP3LEV3</td>
<td>DEP3LEV3</td>
<td>PTF</td>
<td>8</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>DEP3LEV3</td>
<td>PTB</td>
<td>8</td>
<td>1</td>
</tr>
<tr>
<td>DEP4LEV2</td>
<td>DEP4LEV2</td>
<td>PTF</td>
<td>8</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>DEP4LEV2</td>
<td>PTB</td>
<td>8</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>DEP5LEV3</td>
<td>PCF</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>DEP6LEV3</td>
<td>PCF</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>DEP5LEV3</td>
<td>DEP5LEV3</td>
<td>PTF</td>
<td>8</td>
<td>1</td>
</tr>
<tr>
<td>DEP6LEV3</td>
<td>DEP6LEV3</td>
<td>PTF</td>
<td>8</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>DEP6LEV3</td>
<td>PTB</td>
<td>8</td>
<td>3</td>
</tr>
</tbody>
</table>

Figure 14. Pointer Count Summary report (for an HDAM database)
The Pointer Count Summary report also provides the following information:

- If the given segment type has PTF pointers, you can obtain the number of twin chains for a given segment type by looking in the SOURCE SEGMENT field and the NBR OF SOURCE SEGMENTS ZERO PTR field for the given segment type. If the given segment type has hierarchical pointers, but its parent has a PCF pointer pointing to the given segment type, you can also obtain this number by looking in the TARGET SEGMENT, PTR TYPE, and NBR OF SOURCE SEGMENTS WITH PTR fields.

- You can obtain the average number of twin chain occurrences for a given segment type by using the number of twin chains just described. After determining the number of twin chains, divide that number into the total number of source segments (ZERO PTR + WITH PTR) for the segment type.

**Pointer Statistics report**

This report contains a list of all the pointers to a user-specified target, and various totals and ratios for each segment pointer type.

The following figure shows an example of the Pointer Statistics report for an HDAM database.

---

<table>
<thead>
<tr>
<th>SOURCE SEGMENT</th>
<th>TARGET SEGMENT</th>
<th>PTR TYPE</th>
<th>NBR OF PTRS (POINT TO)</th>
<th>AVERAGE BLOCKS AWAY</th>
<th>I/O PROB</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROOTLEV1</td>
<td>ROOTLEV1</td>
<td>PTF</td>
<td>SAME BLOCK</td>
<td>DIFF BLOCK</td>
<td>ONE BLOCK AWAY</td>
</tr>
<tr>
<td></td>
<td>ROOTLEV1</td>
<td>PTB</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>DEP1LEV2</td>
<td>PCF</td>
<td>0</td>
<td>8</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>DEP2LEV2</td>
<td>PCF</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DEP4LEV2</td>
<td>PCF</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEP1LEV2</td>
<td>DEP1LEV2</td>
<td>PTF</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>DEP1LEV2</td>
<td>PTB</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DEP2LEV2</td>
<td>DEP2LEV2</td>
<td>PTF</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>DEP3LEV3</td>
<td>PCF</td>
<td>8</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>DEP3LEV3</td>
<td>PCL</td>
<td>7</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>DEP3LEV3</td>
<td>DEP3LEV3</td>
<td>PTF</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>DEP3LEV3</td>
<td>PTB</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>DEP4LEV2</td>
<td>DEP4LEV2</td>
<td>PTF</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>DEP4LEV2</td>
<td>PTB</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>DEP5LEV3</td>
<td>PCF</td>
<td>7</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>DEP6LEV3</td>
<td>PCF</td>
<td>6</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>DEP5LEV3</td>
<td>DEP5LEV3</td>
<td>PTF</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>DEP6LEV3</td>
<td>DEP6LEV3</td>
<td>PTF</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>DEP6LEV3</td>
<td>PTB</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

---

*Figure 15. Pointer Statistics report*

The following list describes the fields in the Pointer Statistics report. All values in the report are decimal.
**SOURCE SEGMENT**  
The segment name that has one or more pointers in its prefix.

**TARGET SEGMENT**  
The segment name that is pointed to by the source segment.

**PTR TYPE**  
The type of pointer used. For each pointer in a prefix, a line of information is printed. If the pointer type is HF or HB, the TARGET SEGMENT field contains an asterisk (*).

**NBR OF PTRS (POINT TO) SAME BLOCK**  
The number of pointers of the given type that point to segments in the same physical block or CI.

**NBR OF PTRS (POINT TO) DIFF BLOCK**  
The number of pointers of the given type that point to segments in a different block or CI.

**NBR OF PTRS (POINT TO) ONE BLOCK AWAY**  
A refinement of the NBR OF PTRS (POINT TO) DIFF BLOCK data. For those pointers that point to segments in different blocks, this is the number of pointers that point to the next or previous block or CI.

**AVERAGE BLOCKS AWAY**  
A refinement of the NBR OF PTRS (POINT TO) DIFF BLOCK data. For those pointers that point to segments in different blocks, this is the average number of blocks away. You can manually translate this number into tracks and cylinders by knowing the number of blocks or CIs per track. This data can be used to estimate how far apart the source and target segments are. The larger this number is, the longer the seeks will be.

**I/O PROB**  
The probability of doing I/O in going from the source segment to the target segment using the indicated pointer type. This factor is very important in doing database performance analysis. You can use the I/O PROB information to estimate the number of I/Os needed to retrieve a given segment or sequence of segments.

**Notes:**
- If the source and target segments are in different data set groups, the columns associated with NBR OF PTRS and AVERAGE BLOCKS AWAY are left blank because the data would be meaningless.
- If hierarchical pointers are used, the target segment name is not printed because hierarchical pointers can point to different segment types.

**HALDB Process Summary report**  
This report contains status information of each partition, which is identified by its high key value.

As an example, the following figure shows a HALDB Process Summary report for a PHDAM database.
Figure 16. HALDB Process Summary report

**DBDNAME**
The DBD name of the database.

**DATABASE ORGANIZATION**
Database organization of the processed database. PHDAM or PHIDAM.

**ACCESS METHOD**
Access method of the database.

**NUMBER OF PARTITIONS DEFINED**
The number of partitions that is defined in the HALDB.

**NUMBER OF PARTITIONS PROCESSED**
The number of partitions that is processed in the HALDB.

**PARTITION SELECTION EXIT**
The name of the partition selection exit, if used. When the high key selection is used, this field shows “N/A”.

**P**
Partition status after reload, which can be one of the following characters:

- **I**  Initialized
- **R**  Reloaded
- **Blank**  Not processed

**SEQ**
Sequence number

**NAME**
The partition name that is defined by the IMS HALDB Partition Definition utility.

**ID**
The partition ID.

**PARTITION HIGH KEY**
Value of the high key. This field is not shown when the partition selection exit is used.
PARTITION SELECTION STRING
Value of the partition selection string. This field is not shown when the high key selection is used.

**GDG Base Information report**
This report contains the messages that were issued while creating a GDG base entry.

The following figure shows an example of the GDG Base Information report.

---

**WFP Process Summary report**
This report summarizes information about the WFP data sets that are created in the job.

The following figure shows an example of the WFP Process Summary report.
The following list describes each item in the WFP Process Summary report:

**P**  
The status of the WFP data sets. Either I (initialized) or L (loaded) is shown.

**PARTITION**  
The partition name of the database.

**DDNAME**  
The DD name of the WFP data set.

**DSNAME**  
The name of the WFP data set.

**PSINDEX**  
The name of the partition secondary index database.

**INDEX ENTRIES**  
The number of index pointer segment entries in the WFP data set.

**SNAP data set**  
The SNAP data set contains the DBD Snap Dump report.

**DBD Snap Dump report**  
The DBD Snap Dump report contains a snapshot of the DBD when the DBDSNAP=YES statement is specified in the FRRIN data set.

When a snap is taken, the DBD and some of the IMS HP Load control blocks are provided. The snap gives the virtual storage addresses of the data traced on the right. There might be multiple lines, each showing up to 32 bytes of data in the hexadecimal and EBCDIC format.

The following figure shows an example of the DBD Snap Dump report.

**HPSRAMSO data set**  
The HPSRAMSO data set contains the Result of Database Data Set Allocation report.
Result of Database Data Set Allocation report

This report contains results from the IDCAMS commands that were issued to allocate database data sets.

The following figure shows an example of the Result of Database Data Set Allocation report.
*** MEMBER NAME: PHIV01AA ***

DELETE (DBSAMPO1.RELOAD.DATA1.A00001) PURGE CLUSTER
IDC0505I ENTRY (D) DBSAMPO1.RELOAD.DATA1.A00001.DATA DELETED
IDC0505I ENTRY (C) DBSAMPO1.RELOAD.DATA1.A00001 DELETED
IDC0001I FUNCTION COMPLETED, HIGHEST CONDITION CODE WAS 0

IF LASTCC<=8 THEN MAXCC=0

DEFINE CLUSTER (NAME(DBSAMPO1.RELOAD.DATA1.A00001) -
    VOL(+) NONINDEXED REUSE SHR(3 3) -
    TRK(1 1) CISZ(4096) RECSZ(4089 4089))
IDC0508I DATA ALLOCATION STATUS FOR VOLUME TST057 IS 0
IDC0512I NAME GENERATED-(D) DBSAMPO1.RELOAD.DATA1.A00001.DATA
IDC0181I STORAGECLASS USED IS SCIPRMV
IDC0001I FUNCTION COMPLETED, HIGHEST CONDITION CODE WAS 0

IDC0002I IDCAMS PROCESSING COMPLETE. MAXIMUM CONDITION CODE WAS 0

*** MEMBER NAME: PHIV01AL ***

DELETE (DBSAMPO1.RELOAD.DATA1.L00001) PURGE CLUSTER
IDC0505I ENTRY (D) DBSAMPO1.RELOAD.DATA1.L00001.DATA DELETED
IDC0505I ENTRY (I) DBSAMPO1.RELOAD.DATA1.L00001.INDEX DELETED
IDC0505I ENTRY (C) DBSAMPO1.RELOAD.DATA1.L00001 DELETED
IDC0001I FUNCTION COMPLETED, HIGHEST CONDITION CODE WAS 0

SET MAXCC=0

DEFINE CLUSTER (NAME(DBSAMPO1.RELOAD.DATA1.L00001) -
    VOL(+) INDEXED SHR(3 3) -
    KEYS(9,0) -
    TRK(1 1) CISZ(512) RECSZ(50 50))
IDC0508I DATA ALLOCATION STATUS FOR VOLUME TST064 IS 0
IDC0509I INDEX ALLOCATION STATUS FOR VOLUME TST064 IS 0
IDC0512I NAME GENERATED-(D) DBSAMPO1.RELOAD.DATA1.L00001.DATA
IDC0512I NAME GENERATED-(I) DBSAMPO1.RELOAD.DATA1.L00001.INDEX
IDC0181I STORAGECLASS USED IS SCIPRMV
IDC0001I FUNCTION COMPLETED, HIGHEST CONDITION CODE WAS 0

IDC0002I IDCAMS PROCESSING COMPLETE. MAXIMUM CONDITION CODE WAS 0

*** MEMBER NAME: PHIV01AX ***

DELETE (DBSAMPO1.RELOAD.DATA1.X00001) PURGE CLUSTER
IDC0505I ENTRY (D) DBSAMPO1.RELOAD.DATA1.X00001.DATA DELETED
IDC0505I ENTRY (I) DBSAMPO1.RELOAD.DATA1.X00001.INDEX DELETED
IDC0505I ENTRY (C) DBSAMPO1.RELOAD.DATA1.X00001 DELETED
IDC0001I FUNCTION COMPLETED, HIGHEST CONDITION CODE WAS 0

SET MAXCC=0

DEFINE CLUSTER (NAME(DBSAMPO1.RELOAD.DATA1.X00001) -
    VOL(+) INDEXED SHR(3 3) -
    KEYS(8,5) REUSE -
    TRK(1 1) CISZ(512) RECSZ(14 14))
IDC0508I DATA ALLOCATION STATUS FOR VOLUME TST062 IS 0
IDC0509I INDEX ALLOCATION STATUS FOR VOLUME TST062 IS 0
IDC0512I NAME GENERATED-(D) DBSAMPO1.RELOAD.DATA1.X00001.DATA
IDC0512I NAME GENERATED-(I) DBSAMPO1.RELOAD.DATA1.X00001.INDEX
IDC0181I STORAGECLASS USED IS SCIPRMV
IDC0001I FUNCTION COMPLETED, HIGHEST CONDITION CODE WAS 0

IDC0002I IDCAMS PROCESSING COMPLETE. MAXIMUM CONDITION CODE WAS 0

Figure 20. Result of Database Data Set Allocation report
Chapter 4. Reorganizing a HALDB database

To reorganize a HALDB database, you must run several programs. Reorganizing a HALDB database, however, is simpler than reorganizing a non-HALDB database because neither complicated job steps for logical relationships nor secondary indexes are required for a HALDB database.

Topics:

- “Scenario: Reorganizing a HALDB database” on page 108
- “Tips for reorganizing partitions individually” on page 110
Scenario: Reorganizing a HALDB database

This scenario demonstrates how to reorganize a HALDB database.

About this task

The following procedure introduces a typical job stream for reorganizing a HALDB database. Some steps are optional.

Tip: If you use IMS HP Unload to unload the database and IMS HP Load to reload the database, you can omit some of the steps in this scenario. For the steps you can omit and for ICL examples to run IMS HP Unload and IMS HP Load, see "Example 3: Reorganizing a HALDB database by using IMS HP Unload and IMS HP Load" on page 131.

Procedure

1. Optional: Back up the partitions you are reorganizing.
   The IMS Database Image Copy utility (DFSUDMP0) or IMS High Performance Image Copy for z/OS creates a copy of each data set group of the old partitions. This is the backup copy of your database. It can be used when something goes wrong during the reorganization process.

   **Recommendation:** This step is not mandatory, however, it is recommended that you always include it in your reorganization job.

2. Unload the partitions.
   The IMS HD Reorganization Unload utility (DFSURGU0) or IMS High Performance Unload creates a sequential data set that contains the old unloaded partitions. You can choose to unload a single partition, a sequence of partitions, or all of the partitions in your database.

3. Optional: Do DBDGEN.
   This IMS procedure creates the DBDs for the reorganized database. If you want to change the parameters in your DBD that affect all partitions, you must reorganize all of the partitions in your database.

4. Optional: Change partition definitions.
   IMS HALDB Partition Definition utility (DFSHALDB) changes the HALDB partition definitions stored in the DBRC RECON data sets.

5. Optional: Delete then define the partition data sets.
   IDCAMS deletes and allocates the partition data sets. For VSAM, you must use the REUSE parameter in your original CLUSTER definition to make this step optional. To switch to the VSAM load mode to improve the performance of the ILDS rebuilding process, specify the REUSE parameter in your ILDS CLUSTER definition, and specify ILDSBLD=YES.

6. Optional: Mark partitions as needing initialization.
   The DBRC CHANGE.DB command marks a partition as needing initialization. If you are sure that every partition you are reorganizing will be populated by segments after reload, you can skip this step. If not, mark all partitions you are reorganizing as needing initialization.

7. Run the Physical Sequence Sort for Reload utility.
   If you change the partition boundaries during the reorganization, this step is mandatory. Physical Sequence Sort for Reload sorts the unloaded database records so that the partitions are in the order that is required for the subsequent IMS HP Load run.
8. Run IMS HP Load.
   If you mark the partitions as needing initialization, specify PARTINIT=YES in your IMS HP Load job.

9. Rebuild the indirect list data set (ILDS).
   Rebuild the ILDS by using the IMS HALDB Index/ILDS Rebuild utility (DFSPREC0). This utility creates the ILDS from the HALDB partition data.
   If the ILDS is already rebuilt in the IMS HP Load step, you can omit this step.

10. Back up the partitions you have reorganized.
    This step creates the first backup copy of the new partitions.
Tips for reorganizing partitions individually

If you choose to reorganize the partitions individually, you can achieve a degree of parallelism in your reorganization process because all of the partition reorganization jobs can run simultaneously. You can reduce the overall reorganization time by running your jobs simultaneously.

In this case, however, you cannot use the partition initialization function during the reload processing that is done by IMS HP Load because other partitions might be being reorganized simultaneously. You need to initialize the partition before the reload processing, and specify PARTINIT=NO in each IMS HP Load job. For the partition initialization processing, you can use IMS HP Load job by specifying a dummy DFSUINPT DD in place of the IMS Database Prereorganization utility (DFSURPR0).
Chapter 5. Reloading user-defined header format records

You can reload database records from an unloaded file that has user-defined header format.

IMS HP Load provides the following two methods to reload such records:

**Specifying each element of the header portion**
You can specify parameters on the USERHDR control statement according to the contents in the user-defined header format records of the unloaded file.

**Using a header conversion routine**
You can provide a user header conversion exit routine, convert the user header record to the IMS HD format, and specify the name of the exit routine in the USERHDRX control statement.

The user header conversion routine is intended to be used mainly by system programmers who have good knowledge of IMS database, IMS HP Load, and IMS HP Unload.

Topics:
- "Restrictions for reloading user-defined header format records" on page 112
- "Programming interface of user header conversion routine" on page 113

Related reference:
- “USERHDR control statement” on page 73
- “USERHDRX control statement” on page 75
Restrictions for reloading user-defined header format records

Certain restrictions apply when you reload records that have user-defined header format.

- If you specify the USERHDRX control statement and the SORT control statement at the same time, you must specify one of the following exit routine names on the USERHDRX control statement.

<table>
<thead>
<tr>
<th>Exit routine name</th>
<th>Format of the unloaded file</th>
</tr>
</thead>
<tbody>
<tr>
<td>HPSRCVA3</td>
<td>*A3</td>
</tr>
<tr>
<td>HPSRCVA4</td>
<td>*A4</td>
</tr>
<tr>
<td>HPSRCVF1</td>
<td>*F1</td>
</tr>
<tr>
<td>HPSRCVF2</td>
<td>*F2</td>
</tr>
<tr>
<td>HPSRCVF3</td>
<td>*F3</td>
</tr>
<tr>
<td>HPSRCVF6</td>
<td>*F6</td>
</tr>
</tbody>
</table>

- If the segment data of unloaded file is in a compressed form, you must specify COMPRESS=NO.
- HALDB is not supported.
- Logical relationships are not supported.

**Important:** If the database has logical relationships, you must use USERHDRX= at your own risk.

**Requirement:** If the unload data set has logical child segments, the unload header, which is generated by the user header conversion exit, must include enough information to resolve logical relationships.

- If the USERHDR control statement is specified, a header or a trailer record must not reside in the unloaded file.
- The Load API is not supported.
Programming interface of user header conversion routine

Use this reference information to create a user header conversion routine.

This topic describes the interface of the IMS HP Load facilities available to system programmers. It is a product-sensitive programming interface. See the Programming interface information section at the end of this publication to understand the restrictions that are associated with this type of material.

The user exit routine is invoked after every reading of segment data from the unloaded data set that has a user-defined header. You should use the user header conversion exit routine to convert a user header record to the IMS HD format.

When the user header conversion routine is specified, IMS HP Load calls the exit routine before reading the first record from the unloaded data set (INIT call). The user header conversion exit must set the length of the user header at the INIT call.

When IMS HP Load starts to read the record from the unloaded data set, the exit routine is invoked after every reading of segment data.

The user exit routine must be coded in Assembler.

The user exit routine must not change the value of the key record.

The user exit routine must be AMODE=31 or RMODE=24.

The library in which the user exit routine is stored must be concatenated to the STEPLIB, JOBLIB, or LINKLIB DD statement.

Subsections:
- "Register contents upon entry to the user exit"
- "Register contents returned from the user header conversion exit" on page 114

Register contents upon entry to the user exit

The following table lists register contents upon entry into a user header conversion exit.

<table>
<thead>
<tr>
<th>Register</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Parameter list pointer</td>
</tr>
<tr>
<td>13</td>
<td>Address of Save Area</td>
</tr>
<tr>
<td>14</td>
<td>Return address</td>
</tr>
<tr>
<td>15</td>
<td>Entry point address of the user exit</td>
</tr>
</tbody>
</table>

The following table summarizes parameter list pointed to by register 1 upon entry into a user header conversion exit.
Table 13. Parameter list pointed to by register 1 upon entry into a user header conversion exit

<table>
<thead>
<tr>
<th>Word</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Address of DBD name [Note]</td>
</tr>
<tr>
<td>2</td>
<td>Address of DBD [Note]</td>
</tr>
<tr>
<td>3</td>
<td>Address of user header field [Note]</td>
</tr>
<tr>
<td>4</td>
<td>Address of HD record header field [Note]</td>
</tr>
<tr>
<td>5</td>
<td>User header information</td>
</tr>
<tr>
<td></td>
<td>Byte 1–2: Length of user header</td>
</tr>
<tr>
<td></td>
<td>Byte 3–4: Flag that indicates whether data is compressed or not</td>
</tr>
</tbody>
</table>

**Note:** The address is NULL when the user exit is called for the first time (INIT call).

A(DBD name)
Address of an 8-byte area that contains the DBD name of the database to be processed. When the INIT is called, this field is null.

A(DBD)
Address of an 8-byte area that contains the DBD name of the database to be processed. When the INIT is called, this field is null.

A(User header field)
Address of a starting point of user header field. When the INIT is called, this field is null.

A(HD record header field)
Address of a starting point of HD record header field. When the INIT is called, this field is null.

User header information

**Byte 1–2**
This 2-byte field contains the length of the user header. The RDW length is excluded.

**Byte 3–4**
This 2-byte field contains the following flags:
- `X'8000'`
  - The data is compressed.
- `X'4000'`
  - Unload data set has a header or a trailer record.
- `X'0000'`
  - The data is not compressed.

Register contents returned from the user header conversion exit

The following table shows register contents returned from the user exit routine:

Table 14. Register contents returned from a user exit

<table>
<thead>
<tr>
<th>Register</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>Return code</td>
</tr>
</tbody>
</table>

At the exit point of the user header conversion exit routine, register 15 must contain one of the return codes shown in the following table.
Table 15. Return codes from the user header conversion exit

<table>
<thead>
<tr>
<th>Return code</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Successful completion. IMS HP Load uses the segment data pointed to by the second word of the parameter list.</td>
</tr>
<tr>
<td>4</td>
<td>Skip the header or the trailer record. When the user header conversion exit gets the header or the trailer record, a return code of 4 should be returned.</td>
</tr>
<tr>
<td>Others</td>
<td>Force to terminate the Load process. When IMS HP Load gets a nonzero return code from the user header conversion routine, IMS HP Load terminates with abend U3717.</td>
</tr>
</tbody>
</table>
Chapter 6. Reloading a corrupted database

You can use IMS High Performance Unload and IMS High Performance Load to unload and reload only the correct segments from a corrupted database.

By using IMS High Performance Unload for z/OS (5655-E06) with the SKIPERROR and the DIAGG options, you can unload a corrupted database by skipping incorrect pointers.

For details about unloading a corrupted database, see the topic about unloading corrupted databases in the *IMS High Performance Unload User’s Guide*.

When reloading a database by using an unloaded data set that is created from a corrupted database, you might get a user abend from your IMS HP Load job because duplicated keys remain in the unloaded data set. The following subsections describe how to avoid the user abend in such a case.

Subsections:
- “Specifying SORTOUTD DD for Physical Sequence Sort for Reload”
- “Specifying SEQERROR=SAVE parameter for IMS HP Load”

**Specifying SORTOUTD DD for Physical Sequence Sort for Reload**

You can use the Physical Sequence Sort for Reload utility (PSSR) to check the duplication of keys in HDAM root segments. When the SORTOUTD DD is specified in your PSSR job, database records with root key duplication are placed in the SORTOUTD data set. You can review the stored database records to determine which database record to use for the subsequent reload job.

For more information, see the explanation of SORTOUTD DD in “PSSR JCL requirements” on page 170.

**Specifying SEQERROR=SAVE parameter for IMS HP Load**

You can use the SEQERROR=SAVE parameter and the SEQERROR DD for your IMS HP Load job to check the duplication of keys in all dependent segments and HIDAM root segments. When this parameter is specified, segments with a key duplication and their dependent segments are all placed in the SEQERROR DD data set, and the reload processing continues.

For more information, see “SEQERROR control statement” on page 69.
Chapter 7. User exit interface

By providing a user exit routine, you can perform additional processing or editing of segments during the database reload process.

This topic describes the interface of the IMS HP Load facilities available to system programmers. It is a product-sensitive programming interface. See the Programming interface information section at the end of this publication to understand the restrictions that are associated with this type of material.

If you want to perform additional processing or editing of segments during the database reload, you can provide your own user exit routine. You can code these routines in Assembler, COBOL, or PL/I language. The routine must be link-edited with the REUSE option.

The user exit routine is invoked after every reading of segment data from the HD unloaded data set. You can use the user exit routine to check, modify, delete, and control the placement of segments.

If the segment data is compressed with the DECOMPRESS=NO option specification in the IMS HPUnload job, IMS HP Load calls the segment edit/compression exit routine to decompress the segment data before calling the user exit routine, and calls the segment edit/compression exit routine to compress the segment data after calling the user exit routine. This decompression and compression are performed even if COMPRESS=NO is specified in the FRRIN data set.

If the user exit is written in COBOL, or PL/I, it must be run in the Language Environment (LE). You must specify "L" on the INPUT control statement in the FRRIN data set so that IMS HP Load establishes the LE environment by using the preinitialization services. The user exit is invoked by the CEEPIPI call_sub function.

The user exit routine can be either AMODE=24 or 31. For the Load API process, the exit routine must be AMODE=31.

The library in which the user exit routine is stored must be concatenated to the STEPLIB DD statement.

Sample user exits

The following sample user exits are available in the HPS.SHPSSAMP data set:

HPSRXAA0:
A sample user exit that is written in the Assembler language.

HPSRXCC1:
A sample user exit that is written in Enterprise COBOL for z/OS.

HPSRXPP1:
A sample user exit that is written in Enterprise PL/I for z/OS.
Topics:
- “Parameters for the user exit interface” on page 121
- “Interface of Assembler user exit routine” on page 124
- “Interface of Enterprise COBOL for z/OS program user exit routine” on page 125
- “Interface of Enterprise PL/I for z/OS user exit routine” on page 126
Parameters for the user exit interface

The following information describes the parameters for the user exit interface and register contents upon entry and return from the user exit.

This topic describes the interface of the IMS HP Load facilities available to system programmers. It is a product-sensitive programming interface. See the Programming interface information section at the end of this publication to understand the restrictions that are associated with this type of material.

Subsections:
- “Register contents upon entry to the user exit”
- “Register contents returned from the user exit” on page 122

Register contents upon entry to the user exit

The following tables show register contents upon entry to the user exit routine:

Table 16. Register contents upon entry to a user exit

<table>
<thead>
<tr>
<th>Register</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Parameter list pointer</td>
</tr>
<tr>
<td>13</td>
<td>Register save area pointer</td>
</tr>
<tr>
<td>14</td>
<td>Return address</td>
</tr>
<tr>
<td>15</td>
<td>Entry point address of the user exit</td>
</tr>
</tbody>
</table>

Table 17. Parameter list pointed to by register 1 upon entry to a user exit

<table>
<thead>
<tr>
<th>Word</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Address of segment name</td>
</tr>
<tr>
<td>2</td>
<td>Address of segment data</td>
</tr>
<tr>
<td>3</td>
<td>Address of work area</td>
</tr>
<tr>
<td>4</td>
<td>Address of DBD name</td>
</tr>
</tbody>
</table>

A(segment name)
Address of a 9-byte area that contains an 8-byte segment name and a 1-byted segment code in hexadecimal. At the end of the user exit routine, the first 8 bytes of the area contain blanks.

A(segment data)
Address of the segment data read from the HD unloaded data set. IMS HP Load always decompresses compressed data before presenting it to the user exit routine.

A(work area)
Address of a segment work area that is large enough to hold the largest possible segment defined in the DBD being used. If a segment length change is required, this area must be used.

The size must be larger than the maximum segment length and smaller than the maximum segment length plus 1024 bytes.
For variable length segments, you must change the segment length (LL field) by the user exit routine to an appropriate size of the segment being modified. The segment length must not exceed the maximum segment length defined in the DBD being used. The amount of segment data moved into the database data set is controlled by the LL field.

For fixed-length segments, the amount of segment data moved into the database data set is controlled by the segment length defined in the DBD. It is your responsibility to initialize the expanded portion of the segment by the user exit routine if the segment length in the DBD has been expanded. If a return code of zero is returned from the user exit routine, IMS HP Load moves unchanged segment data into the database data set, and automatically initializes the expanded portion of the segment with binary zeros.

When changing the value of the sequence field by using a user exit, the sequence of the field itself cannot be changed.

A(DBD name)
Address of an 8-byte area that contains the DBD name of the database to be processed.

Register contents returned from the user exit

The following table shows register contents returned from the user exit routine:

Table 18. Register contents upon return from a user exit

<table>
<thead>
<tr>
<th>Register</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>Return code</td>
</tr>
</tbody>
</table>

At the exit point of the user exit routine, register 15 must contain the return code so that IMS HP Load can determine the correct process of the segment data. The return code must be one of the codes shown in the following table:

Table 19. Return codes from the user exit

<table>
<thead>
<tr>
<th>Return code</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Indicates that no segments are to be modified. IMS HP Load uses the segment data pointed to by the second word of the parameter list.</td>
</tr>
<tr>
<td>4</td>
<td>Indicates that segments are to be modified. IMS HP Load uses the segment data pointed to by the third word of the parameter list.</td>
</tr>
<tr>
<td>8</td>
<td>A segment has been deleted. IMS HP Load skips the data for this segment data. The user exit routine must return the return code 8 for all dependent segments of this segment.</td>
</tr>
<tr>
<td>12</td>
<td>A segment and all subsequent segments, have been deleted. IMS HP Load skips this segment and data for all subsequent segments in the same database record.</td>
</tr>
<tr>
<td>16</td>
<td>Indicates a request to put an HDAM segment into the overflow area. IMS HP Load puts the segment data into the HDAM overflow area. For the root segment, IMS HP Load puts that root segment and all its subsequent segments in the HDAM overflow area.</td>
</tr>
<tr>
<td>20</td>
<td>Indicates a request to put an HIDAM segment into a new block. IMS HP Load puts the segment data into a new block.</td>
</tr>
<tr>
<td>100–999</td>
<td>Indicates termination. IMS HP Load ends the loading process and passes this return code back as the job return code. The available return code value is in the range of 100 - 999.</td>
</tr>
</tbody>
</table>
Interface of Assembler user exit routine

When you use an Assembler user exit routine, ensure that the exit routine meets the following requirements.

This topic contains product-sensitive programming interface information. See the Programming interface information section at the end of this publication to understand the restrictions that are associated with this type of material.

- It must not be specified L after the user exit name in the FRRIN data set.
- The Assembler user exit routine is simply called by IMS HP Load. The Language Environment is not established.

A sample of this user exit is HPSRXAA0 in the HPS.SHPSSAMP data set.
Interface of Enterprise COBOL for z/OS program user exit routine

When you use an Enterprise COBOL for z/OS program user exit routine, ensure that the exit routine meets the following requirements.

This topic contains product-sensitive programming interface information. See the Programming interface information section at the end of this publication to understand the restrictions that are associated with this type of material.

- It must be specified L after the user exit name in the FRRIN data set.
- The user exit routine is called in the Language Environment (LE). LE libraries need to be used as follows:
  - SCEERUN is needed during compiling the user exit and running IMS HP Load.
  - SCEELKED is needed during the linkage-edit process of the user exit.
- The return code must be set in RETURN-CODE.
- STOP RUN cannot be used in the program.

A sample of this user exit is HPSRXCC1 in the HPS.SHPSSAMP data set.
Interface of Enterprise PL/I for z/OS user exit routine

When you use an Enterprise PL/I for z/OS user exit routine, ensure that the exit routine meets the following requirements.

This topic contains product-sensitive programming interface information. See the Programming interface information section at the end of this publication to understand the restrictions that are associated with this type of material.

- It must be specified L after the user exit name in the FRRIN data set.
- The user exit routine is called in the Language Environment (LE). LE libraries need to be used as follows:
  - SCEERUN is needed during compiling the user exit and running IMS HP Load.
  - SCEELKED is needed during the linkage-edit process of the user exit.
- The PL/I user exit routine is loaded dynamically and called by IMS HP Load as a subprogram. It needs to be specified as OPTIONS(FETCHABLE) on the procedure statement in the program. OPTIONS(MAIN) cannot be specified.
- The parameters in the PL/I routine must be declared as pointer variables.
- The PL/I built-in function PLIRETC or LE callable service CEE3SRC can be used to set the return code.
- POSIX(ON) cannot be specified for the LE runtime option.
- Nested or multiple enclaves are not supported. For more information about the nested enclave, see the topic "Using Nested Enclaves" in the Language Environment Programming Guide.

A sample of this user exit is HPSRXPP1 in the HPS.SHPSSAMP data set.
Chapter 8. Examples

The following examples present typical effective uses of IMS HP Load.

Topics:
- “Example 1: As a part of the reorganization process” on page 128
- “Example 2: Creating an empty database” on page 130
- “Example 3: Reorganizing a HALDB database by using IMS HP Unload and IMS HP Load” on page 131
- “Example 4: Creating WFP work data sets to rebuild PSINDEXes” on page 133
Example 1: As a part of the reorganization process

The HDAM or HIDAM database reorganization process with no logical relationship consists of three steps. The execution of IMS HP Load is used for one of these steps.

Subsections:
1. “Database unload step”
2. “Allocation step”
3. “Database reload step (using IMS HP Load)” on page 129

Note: In these steps, DBRC is not being used.

Database unload step

The following JCL example is for unloading a VSAM HDAM database.

```jcl
//**-------------------------------------------------------------------*
//**     IMS HD REORGANIZATION UNLOAD UTILITY                          *
//**-------------------------------------------------------------------*
//UNLOAD EXEC PGM=DFSRRCO0,PARM='ULU,DFSURGU0,DBSAMP01,,,,,,,,,,,N,N'
//STEPLIB DD DSN=IMSESA.RESLIB,DISP=SHR
//DFSRESLB DD DSN=IMSESA.RESLIB,DISP=SHR
//IMS DD DSN=IMSESA.DBDLIB,DISP=SHR
//DFSURGU1 DD DSN=DBSAMP01.UNLOAD,DISP=(NEW,PASS),
//   SPACE=(CLY,(20,10))
//SYSPRNIT DD SYSOUT=A
//SYSUDUMP DD SYSOUT=A
//DBSAMP01 DD DSN=DBSAMP01.DATA1,DISP=OLD
//DFSVSAM DD *
1024,10
2048,10
4096,10
/*
```

Figure 21. Example JCL to unload an HDAM database

Allocation step

The following JCL example is for allocating a new VSAM database data set.
Database reload step (using IMS HP Load)

The following JCL example is for reloading a VSAM HDAM database. This reload step also prints statistical reports.

```jcl
//*-------------------------------------------------------------------*
//* ALLOCATE NEW DATABASE  (DATA : VSAM ESDS)  *
//*-------------------------------------------------------------------*
//ALLOC  EXEC  PGM=IDCAMS
//SYSPRINT  DD  SYSOUT=A
//SYSIN  DD *
DELETE (DBSAMP01.RELOAD.DATA1) CLUSTER
DEFINE CLUSTER -
  (NAME(DBSAMP01.RELOAD.DATA1) NONINDEXED -
    VOL(*) -
    RECORDSIZE (2041,2041) -
    CONTROLINTERVALSIZE (2048) -
    SPEED -
    CYL(20 10))
/*

Figure 22. Example JCL to allocate a new database data set

```jcl
//*-------------------------------------------------------------------*
//* IMS HIGH PERFORMANCE LOAD  *
//*-------------------------------------------------------------------*
//RELOAD  EXEC  PGM=HPSRRC00,PARM='ULU,DFSURGLO,IMSESA,........,N,N'
//STEPLIB  DD  DSN=HPS.SHPSLMD0,DISP=SHR
//  DD  DSN=IMSESA.RESLIB,DISP=SHR
//  DD  DSN=IMSESA.DBDLIB,DISP=SHR
//DFSUINPT  DD  DSN=DBSAMP01.UNLOAD,DISP=(OLD,PASS)
//SYSPRINT  DD  SYSOUT=A
//SYSDUMP  DD  SYSOUT=A
//DBSAMP01  DD  DSN=DBSAMP01.DATA1,DISP=OLD
//DFSVSAM01  DD *
VSRBF=32768,10
IOBF=(32K,10)
/*
//FFRIN  DD *
//SEARCH=-10
/*

Figure 23. Example JCL to reload an HDAM database
Example 2: Creating an empty database

There are occasions when you need an empty database, particularly in the online environment. For example, IMS does not allow you to create an HD-type database online. To circumvent this restriction, you create a database offline by inserting one root segment and then deleting it. This method initializes the database so that it can be used online.

You can have IMS HP Load perform this initialization function by simply modifying the JCL statement. Insertion of the following JCL statement creates a usable empty database:

```
//DFSUINPT DD DUMMY
```

If this is an HDAM database, the entire root addressable area is preformatted. This improves online performance when segments are inserted to this database.

If this is an HIDAM database, the database contains a root segment with a key of all X'FF'.
Example 3: Reorganizing a HALDB database by using IMS HP Unload and IMS HP Load

By using IMS HP Unload and IMS HP Load, you can simplify the steps for reorganizing HALDB databases.

A typical scenario for reorganizing a HALDB database is described in Chapter 4, “Reorganizing a HALDB database,” on page 107. If you use IMS HP Unload and IMS HP Load to reorganize a HALDB database, you can omit some of the steps that are introduced in that scenario.

Use the following JCL example to reorganize a PHDAM database that has some PSINDEXes.

```jcl
//**-------------------------------------------------------------*
//* IMS HIGH PERFORMANCE UNLOAD (FABHURG1)
//**-------------------------------------------------------------*
//HPUL EXEC FABHULU, MBR=FABHURG1, DBD=DBSAMP01, DBRC=Y,
//                     DBTLMD=HPS.SHPSLMD0
//SYSPRINT DD SYSOUT**
//RECON1 DD DISP=SHR, DSN=IMSESA.RECON1
//RECON2 DD DISP=SHR, DSN=IMSESA.RECON2
//RECON3 DD DISP=SHR, DSN=IMSESA.RECON3
//SYSUT2 DD DISP=OLD, DSN=OBSDMP01.DATA1
//SYSIN DD *
CHECKREC YES /*
//**-------------------------------------------------------------*
//* DELETE/DEFINE DBDS AND ILDS
//**-------------------------------------------------------------*
//DEFDB EXEC PGM=IDCAMS
//SYSIN DD *
DELETE (OBSDMP01.RELOAD.A00001) CLUSTER
DELETE (OBSDMP01.RELOAD.L00001) CLUSTER
DELETE (OBSDMP01.RELOAD.A00002) CLUSTER
DELETE (OBSDMP01.RELOAD.L00002) CLUSTER

DEFINE CLUSTER (NAME(OBSDMP01.RELOAD.A00001)) -
       REUSE SPEED NONINDEXED SHR(3 3)
       CISZ(4096) RECSZ(4089 4089)
       VOL(*) CYL(100 100))

DEFINE CLUSTER (NAME(OBSDMP01.RELOAD.L00001)) -
       REUSE INDEXED KEYS(9,0) SHR(3 3)
       CISZ(4096) RECSZ(50 50)
       VOL(*) CYL(10 10))

DEFINE CLUSTER (NAME(OBSDMP01.RELOAD.A00002)) -
       REUSE SPEED NONINDEXED SHR(3 3)
       CISZ(4096) RECSZ(4089 4089)
       VOL(*) CYL(100 100))

DEFINE CLUSTER (NAME(OBSDMP01.RELOAD.L00002)) -
       REUSE INDEXED KEYS(9,0) SHR(3 3)
       CISZ(4096) RECSZ(50 50)
       VOL(*) CYL(10 10))
/*
```

Figure 24. Example JCL to reorganize a PHDAM database by using IMS HP Unload and IMS HP Load (Part 1 of 2)
In this example:

- The CHECKREC YES option is specified for the IMS HP Unload job step. Also, the PARTINIT=YES,UNLOAD option is specified for the IMS HP Load job step. When these two options are specified, you do not need to issue DBRC CHANGE.DB commands for empty partitions (corresponds to Step 6 on page 108 in “Scenario: Reorganizing a HALDB database” on page 108).

- The SORT=YES option is specified for the IMS HP Load job step. When this option is specified, you can omit the Physical Sequence Sort for Reload step (corresponds to Step 7 on page 108 in “Scenario: Reorganizing a HALDB database” on page 108).

- The ILDSBLD=YES,DSPR option is specified for the IMS HP Load job step. When this option is specified, you do not need to rebuild the ILDS after the reorganization (corresponds to Step 9 on page 109 in “Scenario: Reorganizing a HALDB database” on page 108).
Example 4: Creating WFP work data sets to rebuild PSINDEXes

While reloading HALDB databases with IMS HP Load, you can create WFP data sets, which are work data sets for rebuilding PSINDEXes with IMS Index Builder. By using the WFP data sets as input to IMS Index Builder, you can rebuild PSINDEXes without scanning the database.

If the WFPHLQ control statement is specified on the FRRIN DD statement, IMS HP Load generates one WFP data set for each HALDB partition.

The names of the WFP data sets are determined by the parameter that is specified on the WFPHLQ control statement and by the partition names. For example, if you specify WFPHLQ=IMSTOOLS.WORK.HMPORDP1 and if the partition names are PART001 and PART002, the following WFP data sets are created:

IMSTOOLS.WORK.HMPORDP1.PART001.WFP
IMSTOOLS.WORK.HMPORDP1.PART002.WFP

Use the following JCL example to create WFP data sets.

```
/ *--------------------------------------------------------*
/ *            IMS HIGH PERFORMANCE LOAD                  *
/ *--------------------------------------------------------*
//RELOAD EXEC PGM=HPSRRC00,
 //        PARM='ULU,DFSURGL0,HMPORDP1,,,,,,,,,,,Y,N
//STEPLIB DD DISP=SHR,DSN=HPS.SHPSLMD0
//        DD DISP=SHR,DSN=IMSESA.RESLIB
//        DD DISP=SHR,DSN=IMSESA.DBDLIB
//        DD DISP=SHR,DSN=IMSESA.MDALIB
//DFSUINPT DD DISP=(OLD,PASS),DSN=DBSAMP01.UNLOAD
//SYSPRINT DD SYSOUT=A
//SYSUDUMP DD SYSOUT=A
//DFSVSAMPP DD *
VSRBF=32768,10
IOBF=(32K,10)
/*
//FRRIN DD *
WFPHLQ=IMSTOOLS.WORK.HMPORDP1
WFPPVOLCNT=5
PARTINIT=YES,UNLOAD
/*
```

Figure 26. Example JCL to create WFP data sets

In this example, the WFP data sets are allocated dynamically. Therefore, HPSRPnnn DD statements, which define the names of the WFP data sets, are not specified.

The WFPVOLCNT control statement is optional. You can specify an appropriate value depending on the availability of the volumes.

Requirements:

- If you are not sure whether the HALDB database contains empty partitions, use the unloaded data set that is created by IMS HP Unload with the CHECKREC=YES option. Otherwise, WFP data sets are not created for empty partitions.
- When you rebuild PSINDEXes with IMS Index Builder, specify the same HLQ value on the WFPHLQ statement of IIUIN DD.
Part 3. Application programming interface for IMS High Performance Load

IMS High Performance Load provides an application programming interface that improves the performance of load processing of your application programs.

Topics:

- Chapter 9, “Introduction to the Load API,” on page 137
- Chapter 10, “Operating instructions for using the Load API,” on page 139
Chapter 9. Introduction to the Load API

IMS HP Load provides the Load API. You can use your application program as input of the Load API for the initial load processing. Load API runs the initial load processing much faster than that of IMS DL/I.

**Typical benefits**

The use of the Load API, might benefit an installation by:

- Reducing the time and cost for the initial load process.
- The load API function runs the initial load process much faster than that of IMS DL/I.
- Being also applicable to HALDB.
- The calls that use the Application Interface Block (AIB) is supported.
- The call interface of EXEC DLI is supported.
- You can skip the pre-reorganization job step in the case of a HALDB.
- Having flexible control statements.
- Supporting trace output report.
Chapter 10. Operating instructions for using the Load API

Use the following topics to learn how to use the Load API.

Prerequisite: Before you use the Load API, ensure that your environment is configured for the Load API. For more information, see “Configuring the environment to use the Load API” on page 28.

Topics:

- “Supported language and environment for Load API” on page 140
- “Load API restrictions” on page 141
- “Load API data flow” on page 142
- “Load API JCL requirements” on page 143
- “Load API control statements” on page 146
- “Load API output” on page 149
Supported language and environment for Load API

To use the Load API, the language and the environment must meet the requirements of the Load API.

The following languages are supported for the Load API function:
- Assembler Language
- Enterprise COBOL for z/OS
- Enterprise PL/I for z/OS

The following requirements apply to the Load API environment:
- The library of the application must be specified in the STEPLIB DD statement, and its execution environment must be the Language Environment.
- AMODE=24 and 31, RMODE 24 and ANY are supported.

**Prerequisite:** Before you use the Load API, ensure that your environment is configured for the Load API. For more information, see "Configuring the environment to use the Load API" on page 28.
Load API restrictions

Certain restrictions apply to using the Load API.

The following are the restrictions of the Load API function:

- The load program must specify a PCB with PROCOPT=L or LS.
- The Load API processes only ISRT calls.
- You can insert only one database which has a secondary index per job step.
- It is processed only in the DL/I region.
- It does not support the field level sensitivity.
- If an HDAM database does not define a unique sequence field and the insert rules are 'FIRST' or 'HERE', IMS chains the duplicate segments in the reverse order, that is last in first out.
- Only the command codes D and Null are supported. The V command code can be coded but is ignored.
- The status code does not return in the following cases:
  - I/O error
  - Open error
  - HDAM/PHDAM Root Key duplication error.
  - When the randomizer issues RC=04
- Does not support application programs that run under UCF.
- The SEQERROR control statement is ignored. In that case, SEQERROR=ABEND is assumed.
- The specification of USERHDR= and USERHDRX= are ignored.
- You cannot specify SORTHIDAM=YES.
- If HALDB has a secondary index, the process of Index Builder is required after loading to rebuild the secondary index.
- A logical child of HALDB cannot be inserted.
- The Load API does not support IMS Tools Knowledge Base. You cannot store IMS HP Load reports in the output repository of IMS Tools Knowledge Base when the Load API is used.
Load API data flow

The Load API uses the application program as input. The control statements for the Load API process can be specified in the FRRIN data set.

The following figure shows the data flow of the Load API function.

![Diagram of the Load API data flow]

*Figure 27. The general data flow for the Load API function of IMS HP Load*
Load API JCL requirements

To use the Load API, code the EXEC statement and DD statements that satisfy the Load API JCL requirements.

Subsections:
- “EXEC statement and DD statements”
- “JCL example” on page 145

EXEC statement and DD statements

JCL requirements for running the Load API are shown in the following table.

Table 20. DD Statement for the Load API processing of IMS HP Load

<table>
<thead>
<tr>
<th>DDNAME</th>
<th>Use</th>
<th>Format</th>
<th>Need</th>
</tr>
</thead>
<tbody>
<tr>
<td>STEPLIB</td>
<td>Input</td>
<td></td>
<td>Required</td>
</tr>
<tr>
<td>IMS</td>
<td>Input</td>
<td></td>
<td>Required</td>
</tr>
<tr>
<td>DFSRESLB</td>
<td>Input</td>
<td></td>
<td>Optional</td>
</tr>
<tr>
<td>DFSVSAMP</td>
<td>Input</td>
<td></td>
<td>Optional</td>
</tr>
<tr>
<td>FRRIN</td>
<td>Input</td>
<td>LRECL=80</td>
<td>Required</td>
</tr>
<tr>
<td>SYSPRINT</td>
<td>Output</td>
<td>LRECL=81</td>
<td>Required</td>
</tr>
<tr>
<td>SYSOUT</td>
<td>Output</td>
<td>LRECL=121</td>
<td>Optional</td>
</tr>
<tr>
<td>HPSRTRAC</td>
<td>Output</td>
<td>LRECL=133</td>
<td>Optional</td>
</tr>
<tr>
<td>SORWKnN</td>
<td>work data set</td>
<td></td>
<td>Optional</td>
</tr>
</tbody>
</table>

Note: You can specify this data set and adjust the size of work space for the sort process when the SORT=Y control statement is specified.

EXEC Must be specified in the following format:

```
PGM=HPSRRC00,PARM='DLI,loadpgm,loadpsb'
```

Where,

*loadpgm*  
The name of the application.

*loadpsb*  
The name of the PSB.

You can specify program DFSRRC00 if you do not use functions that require authorization for the STEPLIB DD, such as HPIO or COMPAUTH.

STEPLIB DD

To use the Load API, the names of the libraries that contain the load module of IMS HP Load, IMS, and the application must be specified. The name of the library that contains the load module of IMS HP Load must be specified above that of IMS.

If you specify HPIO=YES or COMPAUTH=YES, the load module libraries that are specified on the STEPLIB DD must be APF-authorized.

IMS DD

To use the Load API, the name of the libraries of DBD and PSB that contains DBD and PSB must be specified.
DFSRESLB
Points to an authorized library that contains the IMS SVC modules. For an
IMS HP Load job, IMSESA.RESLIB and any data set that is concatenated to
it on the DFSRESLB DD statement must be authorized through the
Authorized Program Facility (APF). If you do not specify IMSESA.RESLIB
in the STEPLIB DD statement or a library that is concatenated to the
STEPLIB DD statement is not APF-authorized, the DFSRESLB DD
statement is needed.
For more information, see IMS System Definition.

DFSVSAMP
Describes the data set that contains the buffer pool information that is
required by the IMS DL/I buffer handler. This statement is optional. For
more information about the control statement format, see IMS Database
Administration.

Note: When the DFSVSAMP DD card is omitted, the DL/I Buffer Pool is
not allocated. It is necessary in the case of the HALDB initial load.

SYSPRINT DD
Defines the statistics output data set. This statement is required.

SYSOUT DD
Defines the data set for the output message of sort function. If the DD
statement is not coded, it is dynamically allocated to the SYSOUT stream
of the output class that is specified by the MSGCLASS parameter on the
JOB statement. This statement is optional.

FRRIN DD
Defines a required data set that contains control statements that specify the
functions of IMS HP Load. To use the Load API, one of the following
control statements is required:

- LAPIPCB=
- LAPIPCBNO=
- LAPIPDBD=

Control statements USERHDER=, USERHDERX=, and SEQERROR= are
ignored in the Load API processing.

HPSRTRAC DD
Defines an optional output data set to which trace data is written. This
data set contains 133-bytes fixed-length records.
For details, see “TRACE control statement” on page 147.

SORTWKnn DD
These DD statements are for the intermediate storage data sets used by
DFSORT. If SORT=YES is specified, you can define these data sets and
adjust the size of work space that is used for the sort process. You can
specify multiple data sets for it, nn showing its serial number. This
specification is optional. If you do not specify the statement, DFSORT
allocates the data set dynamically with the sizes that are estimated based
on the size of RAA. If the size of RAA is extremely large or extremely
small, specify the SORTWKnn DD explicitly.
For more information about how to code SORTWKnn DD statements, see
the DFSORT Application Programming Guide.
JCL example

The following figure shows a JCL example for the Load API.

//LOADAPI EXEC PGM=DFSRRRC00,PARM='DLI,\texttt{loadpgm,loadpsb},REGION=4096K
//STEPLIB DD DISP=SHR,DSN=HPS.SHPSLMD0
// DD DISP=SHR,DSN=IMS.SDFSRESL
// DD DISP=SHR,DSN=HPS.APPLLIB
//DFSRESLB DD DISP=SHR,DSN=IMS.SDFSRESL
//IMS DD DISP=SHR,DSN=PSBLIB
// DD DISP=SHR,DSN=OB0LIB
//PROCLIB DD DSN=PROCLIB,DISP=SHR
//IEFRDER DD DSN=\&IMSLOG,DISP=(,DELETE,DELETE),
// UNIT=SYSDA,SPACE=(TRK,(1,1)),
// DCB=(RECFM=VB,BLKSIZE=4096,
// LRECL=4092,BUFNO=2)
//SYSUDUMP DD DUMMY
//IMSMON DD DUMMY
//SYSPRINT DD SYSOUT=*  
//FRRIN DD *
LAPIPCB=TGTPCB
SORT=YES
/*

Figure 28. JCL example to use the Load API

Chapter 10. Operating instructions for using the Load API  145
Load API control statements

You can specify the control statements for the Load API in the FRRIN data set in the same manner as are specified in the reload processing.

The control statements in the following table are supported for the Load API.

Table 21. Load API control statements for process control

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted values</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LAPIPCB=</td>
<td>pcbname</td>
<td>(none)</td>
<td>Specifies the PCB name that is used for the Load API</td>
</tr>
<tr>
<td>LAPIPCBNO=</td>
<td>nnnn</td>
<td>(none)</td>
<td>Specifies the PCB number that is used for the Load API</td>
</tr>
<tr>
<td>LAPIPDBD=</td>
<td>dbdname</td>
<td>(none)</td>
<td>Specifies the DBD name that is used for the Load API</td>
</tr>
<tr>
<td>PIPEOWAIT=</td>
<td>time</td>
<td>30</td>
<td>Specifies the time limit for waiting the pipe that is being established</td>
</tr>
<tr>
<td>TRACE=</td>
<td>YES, startnumber, SNAP</td>
<td>NO</td>
<td>Specifies whether to write the trace record</td>
</tr>
</tbody>
</table>

LAPIPCB control statement

Specifies the PCB name that is used for the Load API.

This control statement is optional.

```
>>> LAPIPCB=pcbname
```

pcbname

Specify the name of the PCB in the PSB that is specified on the EXEC statements. PCB name is the label field or the name that is specified in the PCB statement at PSBGEN.

This control statement is mutually exclusive with the LAPIPCBNO and LAPIPDBD control statements.

LAPIPCBNO control statement

Specifies the PCB number that is used for the Load API.

This control statement is optional.
Specify the number of the PCB in the PSB which is specified on the EXEC statements. The number refers to the order of the DBPCBs, and does not contain I/O PCB or the alternate PCB.

The DBPCB with LIST=NO specification is also counted as a number.

This control statement is mutually exclusive with the LAPIPCB and LAPIPDBD control statements.

**LAPIPDBD control statement**

Specifies the DBD name that is used for the Load API.

This control statement is optional.

Specify the name of the DBD to which the PCB used for the Load API refers.

If there are multiple PCBs that refer to the same DBD, the PCB specified in the higher position in the PCB list is used.

This control statement is mutually exclusive with the LAPIPCB and LAPIPDBDNO control statements.

**PIPEOWAIT control statement**

Specifies the time limit, in which the pipe manager stays in the wait state until the session of the pipe connection is established completely. If it exceeds the limit, IMS HP Load terminates abnormally.

Specify the time limit by seconds, in which the pipe manager is in the wait state. You can specify a value in the range of 30 - 3600 (1 hour). If you do not specify this control statement, PIPEOWAIT=30 is used.

**TRACE control statement**

Specify where to write the trace data.
This statement is optional. If you do not specify this control statement, TRACE=NO is used.

YES  Writes the trace records.

StartNumber
    Specifies the starting number of the calls to be traced. If you specify 100, the trace records of the calls from 100th to the last are displayed.

NO   Does not write the trace records.

SNAP  Displays information of function code, I/O area, and SSAs about the last function that was processed in the application program. They are displayed in the Application Abend Snap report when the application ends abnormally.

To define the output location for the trace record, specify HPSRTRAC DD. To write the trace on the output stream, specify as follows:

```
//HPSRTRAC DD SYSOUT=*  
```

To write it on the data set that is specified in HPSRTRAC DD, specify as follows:

```
//HPSRTRAC DD DISP=(NEW,CATLG),DSN=HPLD.SAMPLE01.TRACE,  
//UNIT=SYSDA,SPACE=(CYL,(1,1)),VOL=SER=VOL001,  
//DCB=(DSORG=PS,RECFM=FBA,LRECL=133,BLKSIZE=133)  
```

The required size for the data set depends on the number of calls.
Load API output

The output from the Load API includes status codes, the Trace Output report, and the Application Abend Snap report.

Status codes

The Load API returns status code to user application program for each ISRT call.

The following table shows the status codes that are generated by the Load API for the ISRT calls. The status codes are similar to the status codes of IMS.

*Table 22. Status codes of the Load API*

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC</td>
<td>The specification of segment is incorrect in SSAs. It is not in a hierarchical sequence.</td>
</tr>
<tr>
<td>AH</td>
<td>Required SSA is missing.</td>
</tr>
<tr>
<td>AJ</td>
<td>One of the following errors occurred:</td>
</tr>
<tr>
<td></td>
<td>• Unsupported command code is specified in the PCB</td>
</tr>
<tr>
<td></td>
<td>• Incorrect SSA format was encountered</td>
</tr>
<tr>
<td></td>
<td>• The I/O area length in AIB is incorrect</td>
</tr>
<tr>
<td>AK</td>
<td>Incorrect field name is specified in SSA.</td>
</tr>
<tr>
<td>AM</td>
<td>Only ISRT can be called. The call function is not compatible with the processing option, segment sensitivity, transaction code, definition, or the program type.</td>
</tr>
<tr>
<td>FM</td>
<td>The target partition is not found.</td>
</tr>
<tr>
<td></td>
<td>The key that is supplied on a DL/I call to a HALDB was greater than the high key value for the last partition, or the user’s partition selection exit returned RC=04 after having been passed a key value with which to select a partition.</td>
</tr>
<tr>
<td></td>
<td>This status code is not returned if the randomizer returns RC=04.</td>
</tr>
<tr>
<td>LB</td>
<td>Segment being loaded already exists in the database.</td>
</tr>
<tr>
<td>LC</td>
<td>The key field of segments is out of sequence.</td>
</tr>
<tr>
<td>LD</td>
<td>No parent for this segment has been loaded.</td>
</tr>
<tr>
<td>LE</td>
<td>The hierarchical sequence is not the same as the DBD sequence.</td>
</tr>
<tr>
<td>LF</td>
<td>An attempt was made to load a logical child segment in either a HALDB PHDAM or a PHIDAM database.</td>
</tr>
<tr>
<td>TI</td>
<td>Incorrect system DIB (command-level only). This status code is returned only when the EXEC DLI is used.</td>
</tr>
<tr>
<td>TP</td>
<td>Incorrect number for PCB or incorrect processing option (command-level only). This status code is returned only when the EXEC DLI is used.</td>
</tr>
<tr>
<td>V1</td>
<td>Segment length exceeds the limits of DBDGEN.</td>
</tr>
<tr>
<td>V2</td>
<td>Segment length is not valid (command-level only). This status code is returned only when the EXEC DLI is used.</td>
</tr>
<tr>
<td>V4</td>
<td>Length of variable-length segment is not valid (command-level only). This status code is returned only when the EXEC DLI is used.</td>
</tr>
<tr>
<td>$1</td>
<td>Too many parameters are detected in the call.</td>
</tr>
<tr>
<td>bb</td>
<td>The call was successful.</td>
</tr>
</tbody>
</table>
**Trace Output report**

If you specify TRACE=YES in the FRRIN data set and run, the trace information is produced in the HPSRTRAC DD.

The following figure shows an example of the Trace Output report.

---

**Figure 29. Trace Output report**

**SEGMENT COUNT**
The number of the segment which IMS HP Load processed.

**DBD**
The DBD name of the database referred to by the PCB.

**PCB**
The PCB name.

**PARTITION**
The partition name. (printed only for HALDB)

**SEGMENT NAME**
The segment name.

**SEGMENT CODE**
The segment code.

**SEGMENT DATA**
Data that are returned in the I/O area. This data gives the virtual storage addresses of the data that are traced on the right. The data might consist of multiple lines, each showing 32 bytes of data in hexadecimal and EBCDIC format.

**KEYFEEDB**
Data that are returned in the key feedback area. This data gives the virtual storage addresses of the data that are traced on the right. There might be multiple lines, each showing up to 32 bytes of data in hexadecimal and EBCDIC format.

**Application Abend Snap report**

This report contains information for problem determination of the application program. When the application ends abnormally, this report is produced in the HPSRTRAC DD. It displays information about the last function that was processed in the application program.
The following figure shows an example of the Application Abend Snap report.

Figure 30. Application Abend Snap report

This report includes the following information:

**STATUS CODE**
The status code that is returned in the PCB.

**APPL-PCB**
The PCB that is used by the application program.

**FUNCTION CODE**
The function code that is passed from the application program.

**I/O AREA RETRIEVED LENGTH (HEX)**
The length of the I/O area that is retrieved by IMS HP Load, in hexadecimal format.

**I/O AREA**
The I/O area that is passed from the application program. If the length of the I/O area is longer than 32 K bytes, the first 32 K bytes of data are displayed.

**SSA-nn**
The SSAs that are passed from the application program.

**Notes:**
- FUNCTION CODE, I/O AREA, and SSA-nn are displayed only when the SNAP option is specified on the TRACE control statement.
- If the application program abends by a parameter list error (status AH or $1), or a functional incompatibility (status AM), the Load API function does not display the information about the I/O area or the SSAs.
Part 4. Physical Sequence Sort for Reload

The Physical Sequence Sort for Reload (PSSR) utility sorts an unloaded database data set to reduce the elapsed time that is required to reload the database.

Topics:
- Chapter 11, “Introduction to Physical Sequence Sort for Reload,” on page 155
- Chapter 12, “Sorting an unloaded database data set with PSSR,” on page 161
- Chapter 13, “PSSR user exit,” on page 201
- Chapter 14, “PSSR usage examples,” on page 211
Chapter 11. Introduction to Physical Sequence Sort for Reload

The Physical Sequence Sort for Reload utility (PSSR) sorts an unloaded database data set created from an HDAM, a PHDAM, a HIDAM, a PHIDAM, a HISAM, or a SHISAM database. The utility helps preventing cascading for an HDAM or a PHDAM database and produces reports about database sizes and randomizer performance.

Considerations before using PSSR

The following considerations apply due to the changes made to IMS High Performance Load Version 2 Release 1.

If you reload an HDAM or a PHDAM database by using IMS High Performance Load Version 2 Release 1 or later, the input data must be in the order of hexadecimal value bbbbbbr that represents the result of randomizing module conversion. bbbbb is the block number and rr is the RAP number.

This change was made because direct I/O during the reload process was eliminated to improve performance in IMS High Performance Load Version 2 Release 1. In order to eliminate direct I/O, data must be sorted in sequence before loading occurs. You can specify the SORT control statement in the FRRIN data set to sort the unloaded data in the reload step. Instead of using the SORT control statement, you can run PSSR before reloading. For details about the SORT control statement, see “SORT control statement” on page 70.

The SORTOUT2 data set cannot be used to prevent cascading. The data set cannot be used, by IMS High Performance Load Version 2 Release 1 or later, as an input for reload. To prevent cascading and avoid degradation of performance in reloading, it is recommended that you use the SORT=YES and the SEARCH=-nnn parameter instead of using PSSR. If you reload an HDAM or a PHDAM by using the IMS HD Reorganization Reload utility, you can use the SORTOUT2 data set to prevent cascading.

The following topics provide information about Physical Sequence Sort for Reload.

Topics:
- “Program functions of PSSR” on page 156
- “Typical benefits of PSSR” on page 159
Program functions of PSSR

PSSR sorts an unloaded database data set to prevent cascading and generates reports that summarize database sizes and randomizer performance. The utility also provides support for HALDBs.

Subsections:
• “Sorting an unloaded database data set”
• “Preventing cascading”
• “Providing reports for tuning analysis” on page 157
• “Providing support for HALDB” on page 157

Sorting an unloaded database data set

PSSR is used to sort an unloaded database data set that is produced by any one of the following unload utilities:
• IMS HD Reorganization Unload utility (DFSURGU0)
• Unload utilities of IMS High Performance Unload
• IPR Unload utility of IMS Database Reorganization Expert

The resulting output data set is in physical sequence. The output data set is used as input to the following reload utilities:
• IMS HD Reorganization Reload utility (DFSURGL0)
• IMS HP Load utility of IMS High Performance Load
• IPR Reload utility of IMS Database Reorganization Expert
• IMS System Utilities/Data Base Tools (DBT) Version 2 Fast Reorganization Reload (FRR)

When this output data set is used as input, disk arm movement during the reload process is minimized, which can significantly reduce the elapsed time that is required to reorganize a database. Such a reduction in reload time can often offset the cost of running PSSR, especially when the unloaded database data set is not in physical sequence. (If you change any randomization parameters during the reorganization process, the unloaded database will not be in physical sequence.)

Preventing cascading

PSSR can also be used to help prevent cascading. Cascading might occur when an HDAM database is loaded in physical hierarchical sequence. A database on which cascading has occurred has many database records stored in blocks that are near (but not equal to) their home blocks. Accessing a root segment that is not stored in its home block might require multiple physical input/output operations, which can sometimes cause a degradation of database performance.

When IMS loads a database record (a root segment plus all its dependents), it attempts to put the entire record into its home block. If the home block becomes filled, the remaining segments in that record will spill into nearby blocks. Other records that randomize to a filled block will also spill into nearby blocks.

Loading records in physical sequence can propagate this effect, causing blocks to be filled with records that randomized to previous blocks. As each database record being loaded finds its home block full, it can spill to a higher block. This can cause
the higher block to get full before any of its own root segments are loaded. Such a situation can cascade throughout the loading process.

Whether cascading occurs or not depends on the particular data that is in your database. It can, for example, be caused by a relatively small number of long database records that happen to randomize to the same block.

PSSR helps to avoid the cascading of root segments during reorganization. It does so by loading some database records at the end of the reload, instead of loading them in their normal physical sequence.

Important: If you reload a database by using IMS High Performance Load for z/OS Version 2 Release 1, you cannot use PSSR to prevent cascading. To prevent cascading and avoid degradation of performance in reloading, it is recommended that you use the SORT=YES and the SEARCH=-nnn parameter instead of using PSSR. When the non-zero byte limit value is specified for your database, you must specify ‘NO’ in columns 51-52 of your control statement. Otherwise, IMS HP Load V2 ends abnormally with message HPSR3630E. For more information, see “FABSPSSR CTL data set” on page 178 or “FABSPSSI CTL data set” on page 183.

Providing reports for tuning analysis

PSSR produces several reports that describe database record sizes and randomizer performance. These reports can help you conduct performance and tuning analysis.

Providing support for HALDB

Beside the basic functions for non-HALDB databases, PSSR provides the following functions for HALDBs:

Sorting an unloaded database data set of HALDB

You can apply the sorting function to the entire or selected partitions of database data sets unloaded from a PHDAM or a PHIDAM database. The database data sets are unloaded by the IMS HD Reorganization Unload utility (DFSURGUD0) or the unload utilities of IMS High Performance Unload.

By the use of PSSR, the contents of unloaded database data sets are sorted in the sequence of the partition selections and in physical sequence within a partition. The sequence of partition selection is the sequence in partition high key or another sequence specified by the user-supplied partition selection exit.

By using the unloaded database data set after it is sorted, you can avoid the waste that otherwise occurs while IMS HP Load is reloading the HALDB.

Preventing cascading in each partition of HALDB

PSSR helps preventing cascading within a partition. After sorting as many records into the home block as will fit there, PSSR puts the remaining records to the output data set.

Important: If you reload a database by using IMS High Performance Load for z/OS Version 2 Release 1, you cannot use PSSR to prevent cascading. To prevent cascading and avoid degradation of performance in reloading, it is recommended that you use the SORT=YES and the SEARCH=-nnn parameter instead of using PSSR. When the non-zero byte limit value is specified for your database, you must specify ‘NO’ in columns 51-52 of...
your control statement. Otherwise, IMS HP Load V2 ends abnormally with message HPSR3630E. For more information, see “FABPSSR CTL data set” on page 178 or “FABSPSSI CTL data set” on page 183.

Providing reports of each partition of HALDB
For each partition, PSSR generates not only the usual by-database reports, but also the by-partition reports which include:

- Record Size Distribution report
- Assigned Roots per Block report
- Assigned Roots per RAP report

PSSR also produces a HALDB process summary report, a summary of information about the partition processed.
Typical benefits of PSSR

Physical Sequence Sort for Reload sorts an unloaded database data set that is produced for an HDAM database to reduce the elapsed time that is required to reload the database. It also generates reports that help you conduct performance and tuning analysis.

The PSSR utility offers the following benefits:

**Reduces elapsed time to reload a database**
- Converting database organization to HDAM-type
  - Non-HDAM to HDAM
  - Non-PHDAM to PHDAM
- Converting database organization to HIDAM-type
  - Non-HIDAM to HIDAM
  - Non-PHIDAM to PHIDAM
- Converting database organization to HISAM-type
  - Non-HISAM to HISAM
  - Non-SHISAM to SHISAM
- Changing HDAM randomization parameters (HDAM/PHDAM)
- Changing partition selection exit or High key of HALDB
- Adding or deleting partitions of HALDB

**Helps prevent cascading**

PSSR helps to avoid the cascading of root segments during reorganization. It does so by loading some database records at the end of the reload, instead of loading them in their normal physical sequence.

**Important:** If you reload a database by using IMS High Performance Load for z/OS Version 2 Release 1, you cannot use PSSR to prevent cascading. To prevent cascading and avoid degradation of performance in reloading, it is recommended that you use the SORT=YES and the SEARCH=-nnn parameter instead of using PSSR.

**Helps conduct performance and tuning analysis**

PSSR helps you conduct performance and tuning analysis by generating the following reports:
- Reports about database record sizes
- Reports about DBD parameters
- Reports about randomization
Chapter 12. Sorting an unloaded database data set with PSSR

You can run PSSR to sort an unloaded database data set in the order of physical sequence.

Topics:
- “Restrictions of PSSR” on page 162
- “Program structure and data flow of PSSR” on page 163
- “Reorganizing a database and using PSSR to sort the unloaded data set” on page 166
- “PSSR JCL requirements” on page 170
- “PSSR FABSPSSR input” on page 175
- “PSSR FABSPSSI input” on page 181
- “PSSR output” on page 190
Restrictions of PSSR

Certain restrictions apply to using the PSSR utility.

- PSSR can process an HDAM, a HIDAM, a HISAM, a SHISAM, a PHDAM, or a PHIDAM database.
- The user exit function and the IMS/ESA Year 2000 Exit Tool are supported only when PSSR runs in IMS.
- PSSR does not support any randomizer which refers to the IMS control block, when it runs in non IMS batch.
- PSSR does not support the segment whose key is edited by a Segment Edit/Compression exit.
- If the format of the unloaded file is *CS, PSSR does not support DBD whose segment name is different from DBD used by unload.
Program structure and data flow of PSSR

PSSR runs in a batch environment, either IMS batch environment or non IMS batch environment.

Non IMS batch environment

In non IMS batch environment, the following two DFSORT exit routines are used:

- FABSRT15 is an E15 exit routine that receives control once for each input record before the record is handled by DFSORT. It appends a block number and a root anchor point (RAP) number to each record in the unloaded database data set.
- FABSRT35 is an E35 exit routine that receives control once for each output record after the record is handled by DFSORT. It removes the block number and RAP number (that were appended by FABSRT15) from each record in the sorted unloaded database data set.

PSSR is executed by adding a DFSORT step between the unload and reload steps in a normal database reorganization job.

The following figure shows the general data flow for PSSR in a non IMS batch environment. In addition to supplying input through the SYSIN and CTL (user's requests) data sets, use also the DBDs to control the operation of FABSRT15 and FABSRT35.
IMS batch environment

In IMS batch environment, the PSSR main module FABSMAIN is specified in the PGM= parameter in the EXEC statement. FABSMAIN invokes the DL/I region controller DFSRRC00 and DFSORT program with PSSR sort exit routines.

The following figure shows the general data flow for PSSR in the IMS batch environment.
Figure 32. Data flow for PSSR in IMS batch environment
Reorganizing a database and using PSSR to sort the unloaded data set

You can include a PSSR job step in your reorganization job to sort the unloaded data set during reorganization process.

Before you begin

Several considerations apply for using the output of PSSR as an input of IMS High Performance Load for z/OS Version 2 Release 1 or later. For details, see “Considerations before using PSSR” on page 155.

Procedure

To use PSSR, you must run several programs. These programs can be run in either a single job or in several jobs. A typical PSSR job stream contains some or all of the following steps:

Recommendation: Always include steps 1 and 13 on page 167 in the PSSR job that actually performs a database reorganization. Use of any of the other steps depends on the logical relationships involved.

Note: For information about using IMS utilities, see IMS Database Utilities.

1. Run the IMS Database Image Copy utility (DFSUDMP0) to create a copy of each of the old databases before reorganization.
   These image copies are the backup copies of your databases. These backup copies can be used if something goes wrong in the reorganization process.

2. Run the IMS Database Prereorganization utility (DFSURPR0) to create a control data set (DFSURCDS).
   This control data set is used by the other IMS logical relationship resolution utilities. The DFSURPR0 utility also indicates which databases and segments (if any) need to be scanned by the IMS Database Scan utility.

3. Run the IMS Database Scan utility (DFSURGS0) to scan any databases not being loaded or reorganized for segments that contain logical relationships that are affected by loading or reorganizing other databases.
   For each segment of this type, the utility generates one or more output records, depending upon the relationships in which that segment is involved. It generates an output work data set that will be used by the IMS Database Prefix Resolution utility.

4. Run the IMS HD Reorganization Unload utility (DFSURGU0) or its equivalent program such as unload utilities of IMS High Performance Unload to create a sequential data set that contains the old unloaded databases.

5. Run the DBDGEN procedure.
   This IMS procedure creates the DBDs for the new reorganized databases. You must assemble and link-edit the new DBDs.

6. Run the PSSR utility.
   When PSSR is run in either an IMS environment or a non IMS environment, it sorts the unloaded database data set in physical sequence.

   Note: The only difference between a standard database reorganization job and a database reorganization job that is done by PSSR is the addition of this step. Running a PSSR job includes the following tasks:
a. Select the IBM supplied cataloged procedure to use, or prepare a similar procedure of your own.

There are two ways to run PSSR, and the following two catalog procedures are provided:

**FABSPSSR**

Use the FABSPSSR procedure if you want to run the PSSR job as a standard (non IMS) MVS™ batch job.

**FABSPSSI**

Use the FABSPSSI procedure if you want to run the PSSR job as a batch job using the IMS environment.

For the contents of these cataloged procedures, see "FABSPSSR and FABSPSSI JCL procedures."

b. Code the JCL for the PSSR step.
c. Code the input data for the PSSR programs.
d. Run the PSSR job.
e. Interpret the output reports to verify that the process completed successfully.

7. Run the IDCAMS program to delete the old database data sets and allocate the new database data sets.

8. Run the IMS HD Reorganization Reload utility (DFSURGL0) or an equivalent program such as IMS HP Load to reload a database from the sequential data set that is created by the IMS HD Reorganization Unload utility or its equivalent utility.

This step also creates work data sets that are used as input to the logical relationship resolution utilities.

9. Run the IMS Database Prefix Resolution utility (DFSURG10) to accumulate the information that is generated in work data sets (DFSURCDS and DFSURWF1).

It produces an output data set (DFSURWF3) that contains the prefix information that is needed to complete the logical relationships defined for the databases. It also produces an output data set (DFSURIDX) that contains the information that is needed to create secondary index databases.

10. Run the IMS HISAM ReorganizationUnload utility (DFSURUL0) to format the index work data set (DFSURIDX) to a form that can be used by the IMS HISAM Reorganization Reload utility.

11. Run the IMS HISAM Reorganization Reload utility (DFSURRL0) to create the secondary index databases from the reorganized output data sets provided by the IMS HISAM Reorganization Unload utility.

12. Run the IMS Database Prefix Update utility (DFSURGP0) to update the prefix of each segment of which prefix information is affected by the reload process.

The utility uses the output (DFSURWF3) from the IMS Database Prefix Resolution utility to update the prefix of each segment.

13. Run the IMS Database Image Copy utility (DFSUDMP0) to create a copy of each of the new databases.

These copies are the first backup copies of the new databases.

**FABSPSSR and FABSPSSI JCL procedures**

The following two figures show the contents of the FABSPSSR JCL procedure and the FABSPSSI JCL procedure.
PROC CYL=1,
  DBDLIB='IMSVS.DBDLIB',
  DBTLIB='HPS.SHPSLMD0',
  RNDLIB='IMSVS.RESLIB',
  SRTLIB='SYS1.SORTLIB',
  UNIT=SYSDA,
  SVOL=XXXXXX,
  SUNT=SYSDA,
  IMSHDUN='UNLOAD.FILE'

//*********************************************************************
//* IMS HIGH PERFORMANCE LOAD VERSION 2 (5655-M26)
//* PHYSICAL SEQUENCE SORT FOR RELOAD
//*********************************************************************

//STEPHD EXEC PGM=SORT,REGION=2048K
//STEPLIB DD DSNAME=&SRTLIB,DISP=SHR
//IMS DD DSNAME=&DBDLIB,DISP=SHR
//IMS2 DD DSNAME=&RNDLIB,DISP=SHR
//SORTIN DD DSNAME=&DBTLIB,DISP=SHR
//SORTOUT DD DSNAME=&IMSHDUN,DISP=(OLD,PASS)
//SORTOUT2 DD DSNAME=&UNLSRT2, DISP=(NEW,PASS,DELETE), PART 2 OF 2
//SYSOUT DD SYSOUT**, SORT MESSAGES
//SYSIN DD SYSIN**, CONTAINS 3 OR 4 DFSORT CONTROL STATEMENTS
//CTL DD SYSOUT**, CONTAINS 1 HPS CONTROL STATEMENT

Figure 33. FABSPSSR JCL procedure
//FABSPSSI PROC CYL=1, 00010000
// DBDLIB='IMSVS.DBDLIB', 00020000
// DBTLIB='HPS.SHPSLMD0', 00030000
// RESLIB='IMSVS.RESLIB', 00040000
// RNDLIB='IMSVS.RESLIB', 00050000
// SRTLIB='SYS1.SORTLIB', 00060000
// UNIT=SYSDA, 00070000
// SVOL=XXXXXX, 00080000
// SUNT=3390, 00090000
// IMSHDUN='UNLOAD.FILE' 00100000
// *--------------------------------------------------------------------
// * IMS HIGH PERFORMANCE LOAD VERSION 2 (5655-M26) 00110000
// *--------------------------------------------------------------------
//STEPHD EXEC PGM=FABSMAIN,REGION=4096K 00120000
//STEPLIB DD DSNNAME=&SRTLIB,DISP=SHR 00130000
// DD DSNNAME=&RESLIB,DISP=SHR 00140000
// DD DSNNAME=&DBTLIB,DISP=SHR 00150000
// IMS DD DSNNAME=&DBDLIB,DISP=SHR DBD LOAD MODULE 00160000
//IMS2 DD DSNNAME=&RNDLIB,DISP=SHR 00170000
//DFSRESLB DD DSNNAME=&RESLIB,DISP=SHR 00180000
//SORTIN DD DSNNAME=&IMSHDUN,DISP=(OLD,PASS) INPUT UNLOADED DATA BASE 00190000
//SORTOUT DD DSNNAME=&&UNLSRT,OUTPUT UNLOADED DATA BASE PART 1 OF 2 00200000
// DD DSNNAME=&&UNLSRT2, OUTPUT UNLOADED DATA BASE PART 2 OF 2 00210000
// DD DSNNAME=&UNLSRT, OUTPUT UNLOADED DATA BASE PART 1 OF 2 00220000
// DD DSNNAME=&UNLSRT2, OUTPUT UNLOADED DATA BASE PART 2 OF 2 00230000
// DD DSNNAME=&&UNLSRT, OUTPUT UNLOADED DATA BASE PART 1 OF 2 00240000
// DD DSNNAME=&&UNLSRT2, OUTPUT UNLOADED DATA BASE PART 2 OF 2 00250000
// DD DSNNAME=&&UNLSRT, OUTPUT UNLOADED DATA BASE PART 1 OF 2 00260000
// DD DSNNAME=&&UNLSRT2, OUTPUT UNLOADED DATA BASE PART 2 OF 2 00270000
// DD DSNNAME=&&UNLSRT, OUTPUT UNLOADED DATA BASE PART 1 OF 2 00280000
// DD DSNNAME=&&UNLSRT2, OUTPUT UNLOADED DATA BASE PART 2 OF 2 00290000
// DD DSNNAME=&&UNLSRT, OUTPUT UNLOADED DATA BASE PART 1 OF 2 00300000
// DD DSNNAME=&&UNLSRT2, OUTPUT UNLOADED DATA BASE PART 2 OF 2 00310000
// DD DSNNAME=&&UNLSRT, OUTPUT UNLOADED DATA BASE PART 1 OF 2 00320000
// DD DSNNAME=&&UNLSRT2, OUTPUT UNLOADED DATA BASE PART 2 OF 2 00330000
// DD DSNNAME=&&UNLSRT, OUTPUT UNLOADED DATA BASE PART 1 OF 2 00340000
// DD DSNNAME=&&UNLSRT2, OUTPUT UNLOADED DATA BASE PART 2 OF 2 00350000
// DD DSNNAME=&&UNLSRT, OUTPUT UNLOADED DATA BASE PART 1 OF 2 00360000
// DD DSNNAME=&&UNLSRT2, OUTPUT UNLOADED DATA BASE PART 2 OF 2 00370000
// DD DSNNAME=&&UNLSRT, OUTPUT UNLOADED DATA BASE PART 1 OF 2 00380000
// DD DSNNAME=&&UNLSRT2, OUTPUT UNLOADED DATA BASE PART 2 OF 2 00390000
// DD DSNNAME=&&UNLSRT, OUTPUT UNLOADED DATA BASE PART 1 OF 2 00400000
// DD DSNNAME=&&UNLSRT2, OUTPUT UNLOADED DATA BASE PART 2 OF 2 00410000
// DD DSNNAME=&&UNLSRT, OUTPUT UNLOADED DATA BASE PART 1 OF 2 00420000
// DD DSNNAME=&&UNLSRT2, OUTPUT UNLOADED DATA BASE PART 2 OF 2 00430000
// DD DSNNAME=&&UNLSRT, OUTPUT UNLOADED DATA BASE PART 1 OF 2 00440000
// DD DSNNAME=&&UNLSRT2, OUTPUT UNLOADED DATA BASE PART 2 OF 2 00450000
// DD DSNNAME=&&UNLSRT, OUTPUT UNLOADED DATA BASE PART 1 OF 2 00460000
// DD DSNNAME=&&UNLSRT2, OUTPUT UNLOADED DATA BASE PART 2 OF 2 00470000
// DD DSNNAME=&&UNLSRT, OUTPUT UNLOADED DATA BASE PART 1 OF 2 00480000
// DD DSNNAME=&&UNLSRT2, OUTPUT UNLOADED DATA BASE PART 2 OF 2 00490000
// DD DSNNAME=&&UNLSRT, OUTPUT UNLOADED DATA BASE PART 1 OF 2 00500000
// DD DSNNAME=&&UNLSRT2, OUTPUT UNLOADED DATA BASE PART 2 OF 2 00510000
// DD DSNNAME=&&UNLSRT, OUTPUT UNLOADED DATA BASE PART 1 OF 2 00520000

Figure 34. FABSPSSI JCL procedure

Related reference:
Chapter 14, “PSSR usage examples,” on page 211
To run a PSSR job, code the EXEC statement and DD statements that satisfy the PSSR JCL requirements.

There are two ways to run PSSR:
- As a standard (non IMS) MVS batch job (FABSPSSR procedure)
- As a batch job using the IMS environment (FABSPSSI procedure)

The two procedures require different JCL.

The following table lists the DD statements. Explanations of the JCL parameters follow the table.

### Table 23. DD statements for PSSR

<table>
<thead>
<tr>
<th>DDNAME</th>
<th>Use</th>
<th>Format</th>
<th>Needed in MVS Batch?</th>
<th>Needed in IMS environment?</th>
</tr>
</thead>
<tbody>
<tr>
<td>STEPLIB</td>
<td>Input</td>
<td>PDS</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>SORTIN</td>
<td>Input</td>
<td>RECFM=VB</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>SORCNTL</td>
<td>Input</td>
<td>LRECL=80</td>
<td>Not Applicable</td>
<td>Optional</td>
</tr>
<tr>
<td>SYSIN</td>
<td>Input</td>
<td>LRECL=80</td>
<td>Required</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>CTL</td>
<td>Input</td>
<td>LRECL=80</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>PSSRIN</td>
<td>Input</td>
<td>LRECL=80</td>
<td>Optional</td>
<td>Optional</td>
</tr>
<tr>
<td>IMS</td>
<td>Input</td>
<td>PDS</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>IMS2</td>
<td>Input</td>
<td>PDS</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>SORTOUT</td>
<td>Output</td>
<td></td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>SORTOUT2</td>
<td>Output</td>
<td></td>
<td>Optional</td>
<td>Optional</td>
</tr>
<tr>
<td>SORTOUT3</td>
<td>Output</td>
<td></td>
<td>Optional</td>
<td>Optional</td>
</tr>
<tr>
<td>SORTOUT4</td>
<td>Output</td>
<td></td>
<td>Not Applicable</td>
<td>Optional</td>
</tr>
<tr>
<td>SORTOUTD</td>
<td>Output</td>
<td></td>
<td>Optional</td>
<td>Optional</td>
</tr>
<tr>
<td>SYSPRINT</td>
<td>Output</td>
<td>SYSOUT</td>
<td>Not Applicable</td>
<td>Optional</td>
</tr>
<tr>
<td>SYSOUT</td>
<td>Output</td>
<td>SYSPRINT</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>PR8</td>
<td>Output</td>
<td>LRECL=133</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>PR9</td>
<td>Output</td>
<td>LRECL=133</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>PR9X</td>
<td>Output</td>
<td>LRECL=133</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>PR10</td>
<td>Output</td>
<td>LRECL=133</td>
<td>Not Applicable</td>
<td>Optional</td>
</tr>
<tr>
<td>SYSUDUMP</td>
<td>Output</td>
<td>SYSPRINT</td>
<td>Optional</td>
<td>Optional</td>
</tr>
<tr>
<td>SORTWKMn</td>
<td>Work data set</td>
<td></td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>HDHEADER</td>
<td>Work data set</td>
<td>BLKSIZE=60</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>RECONx</td>
<td>Input/Output</td>
<td></td>
<td>Not Applicable</td>
<td>Optional</td>
</tr>
<tr>
<td>DFSRESLB</td>
<td>Input</td>
<td>PDS</td>
<td>Not Applicable</td>
<td>Optional</td>
</tr>
<tr>
<td>SYSPRINT</td>
<td>Output</td>
<td>SYSPRINT</td>
<td>Not Applicable</td>
<td>Optional</td>
</tr>
</tbody>
</table>

**EXEC**  This statement must have one of the following forms:
- For an MVS batch
For an IMS environment batch

EXEC PGM=FABSMAIN,REGION=xxxxK,PARM=(IMSPLEX=IMSplex_name, DBRCGRP=ccc)

Specify FABSMAIN as the program name. The name is mandatory for the IMS environment. The PARM= parameter is optional, and its keyword are:

**IMSPLEX=IMSplex_name**
Specifies the 1- to 5-character IMSplex name that is used by IMS DBRC SCI Registration.

**DBRCGRP=ccc**
Specifies the 3-character DBRC group identifier that is used by IMS DBRC SCI Registration.

For HALDB processing, an IMS environment batch is required. For non-HALDB processing in an environment where DBRC=FORCE, an IMS environment batch cannot be used.

**STEPLIB DD**
This statement defines the data set that contains the PSSR, DFSORT, and IMS program modules.

If you specify the use of DATXEXIT or a segment edit/compression exit for PSSR, you must include it in this data set.

If the input database is a HALDB, which uses a partition selection exit, you must include the partition selection exit routine in the input RECON definitions in this data set.

**SORTIN DD**
This input data set is the sequential data set that contains the unloaded database. It is created by the IMS HD Reorganization Unload utilities or an equivalent program. The acceptable format of unloaded database data set is the HD, the *SH format, or the communication industry standard format (*CS). The *CS or *SH format data set are created by the unload utilities of IMS High Performance Unload. The record format must be VB no matter whether the data is fixed or variable in length.

**Restriction:** The *CS format is not supported when the input DBD is defined as a HALDB.

**SYSIN DD**
This input data set is required only in the MVS batch. This input data set contains DFSORT control statements. It is not needed in the IMS environment.

**SORTCNTL DD**
This input data set is optional in the IMS environment. It contains DFSORT control statements. It is not needed in the MVS batch.

**CTL DD**
This input data set contains your specification of the processing to be done by PSSR. It specifies the databases that will be reloaded, and it can contain requests you want to make for the database or partitions of HALDB.

**PSSRIN DD**
This input data set contains your specification for an unloaded file format, a user exit, and a DATXEXIT for PSSR.
The user exit and the DATXEXIT are not applicable to the MVS batch.

**IMS DD**
This input data set is a library that contains your DBD load modules. It must contain the DBD of the database being reloaded.

**IMS2 DD**
This input data set is a library that contains the PSSR routines, the user-supplied HDAM randomizing routines, the user-supplied partition selection exit routines that are specified in the CTL control statement, and the user-supplied user exit routines.

When the PSSR job runs under the MVS Batch environment, FABSR15, FABSR35, and other PSSR modules need to be included in the IMS2 data set.

**Requirement:** If you specify any of the partition selection parameters in a PSSR CTL statement, you must include the partition selection exit routine in the IMS2 data set.

**SORTOUT DD**
This output data set is a sequential data set that contains most (or all) of the unloaded database records. Its format must be identical to that of the SORTIN data set. It contains all of the database segments that are intended to be completely loaded into the root addressable area. These database records are sorted into their physical sequence (as it will be after the database is reloaded).

For HALDB, this output data set contains all of the segment records that are not put into output partition data sets. The records are stored in groups of partitions in the order of the partition selection sequence.

The records are stored in a partition in the order of physical sequence.

The partition selection sequence is in high key sequence or in another sequence that is specified by the user-supplied partition selection exit.

**SORTOUT2 DD**
This output data set is the sequential data set that contains all of the unloaded database records that are not in the SORTOUT data set. These records must be in the same format as they were on the SORTIN data set.

For HDAM, this data set cannot be used as an input for the reload processing that is done by IMS High Performance Load for z/OS Version 2 Release 1 or later. For details, see “Considerations before using PSSR” on page 155.

For PHDAM, this data set is used as a work data set for PSSR. You must not reload the segments in this data set. Rather, define this data set as a temporary data set in your PSSR job step when you process HALDB.

For HIDAM or PHIDAM, no data is contained in this data set.

If the SORTOUT3 data set is specified, the database records whose root segment received a return code 4 from the randomizing routine are written into the SORTOUT3 data set instead of this data set.

**SORTOUT3 DD**
This optional data set is a sequential data set that contains database records whose root segment received a return code 4 from the randomizing routine. These records must be in the same format as they were in the SORTIN data set.
Following is an example of this statement for the IBM supplied cataloged procedure that is shown in Figure 33 on page 168.

//SORTOUT3 DD DSNAM=$UNLSRT3,DISP=(NEW,PASS,DELETE),
// SPACE=(CYL,(&CYL)),UNIT=&SUNIT,
// VOLUME=SER=&SVOL,DCB=&IMSHDUN

SORTOUT4 DD
This optional output data set is unique to HALDB. It is a sequential data set that contains database records of root segments that received a nonzero return code from the partition selection. These records must have the same format as in the SORTIN data set.

SORTOUTD DD
This output data set is optional. If it is specified and the root key is defined as unique in the DBD, PSSR checks the root key duplications and stores the duplicated database records into this data set. These records must have the same format as in the SORTIN data set.

Output partition data set DD
This output data set is optional and unique to HALDB. It is a sequential data set that contains all the segment records that must be reloaded into a partition. The name of DD is the partition name that is defined in RECON data sets.

By specifying the DD names for all partitions of your HALDB, you can split your unloaded database data set into a set of partition data sets.

SYSOUT DD
This output data set contains the messages that are produced by DFSORT.

PR8 DD
This output data set contains the reports that are produced by PSSR. The partition information is printed for HALDB. If BLKSIZE is coded on the DD statement, it must be a multiple of 133.

PR9 DD
This output data set contains reports that are produced by PSSR. The partition information is printed for HALDB. If BLKSIZE is coded on the DD statement, it must be a multiple of 133.

PR9X DD
This output data set contains reports that are produced by PSSR. The partition information is printed for HALDB. If BLKSIZE is coded on the DD statement, it must be a multiple of 133.

PR10 DD
This output data set is optional and unique to HALDB. It contains the reports that are produced by PSSR. If BLKSIZE is coded on the DD statement, it must be a multiple of 133.

SYSUDUMP DD (or SYSABEND)
This DD statement defines output from a system abend dump routine. The statement is used only for debugging, when a dump is required.

SORTWKnnn DD
These DD statements are for intermediate storage data sets used by DFSORT. See the DFSORT Application Programming Guide for more information about how to code SORTWKnnn DD statements.

HDHEADER DD
This DD statement is for an intermediate data set that is created by module
FABSRT15 or FABSPE15, and used by module FABSRT35 or FABSPE35. If BLKSIZE is coded on the DD statement, it must be a multiple of 60.

**RECON1 DD, RECON2 DD, RECON3 DD**
These DD statements are for the RECON data sets. The DD statements are required when PSSR processes HALDB and when RECON data sets are not allocated dynamically.

**DFSRESLB DD**
This input data set is required in the IMS environment. It contains the IMS load modules.

**SYSPRINT DD**
This output data set is required in the IMS environment. It contains the messages that are issued by IMS.
PSSR FABSPSSR input

The input data sets differ by the two cataloged procedures. Use the following topics to locate the control statement information for the MVS batch environment (FABSPSSR procedure).

The following input data sets are used in the MVS batch environment, typically by the FABSPSSR procedure:

- SYSIN data set
- CTL data set
- PSSRIN data set

FABSPSSR SYSIN data set

The SYSIN data set contains DFSORT program control statements. These control statements define the type of sort operation to be performed and the sort control fields to be used.

Subsections:
- “Format” on page 176
- “SORT control statement” on page 176
- “MODS control statement” on page 176
- “END control statement” on page 176
- “RECORD control statement” on page 176

Format

This control data set usually resides in the input stream. However, it can be defined as a sequential data set or as a member of a partitioned data set. It usually contains 80-byte fixed-length records. This data set must not be defined as RECFM=U.

The SYSIN data set can be coded as shown in the following example. Three control statements (SORT, MODS, and END) are required. You can also add a RECORD control statement to optimize performance.

Important: Do not change the values underlined in this example.

```
//SYSIN   DD *
SORT FIELDS=(5,20,B1,A),FILSZ=E
MODS E15=(FABSRT15,50000,IMS2,N),E35=(FABSRT35,50000,IMS2,N)
END /*
```

Figure 35. SYSIN data set format

For a HIDAM, a HISAM, or a SHISAM database, specify a SYSIN data set as shown in the following example.

Important: Do not change the values underlined in this example.
//SYSIN  DD
SORT FIELDS=(5,XXX,BI,A),FILSZ=E
MODS 'E15=(FABSRT15,50000,IMS2,N),E35=(FABSRT35,50000,IMS2,N)
END
/

Figure 36. SYSIN data set format for HIDAM, HISAM, and SHISAM databases

Where XXX is KEYLEN + 10.

**SORT control statement**

Describes the control fields (in the input records) on which the program will sort.

The SORT control statement is a required control statement, and must conform to the following rules:
- The FIELDS=(5,20,BI,A) operand is required. It must be coded as shown in [Figure 35 on page 175](#).
- The FIELDS=(5,20,BI,A) operand is required for an HDAM database. For a HIDAM, a HISAM, or a SHISAM database, the FIELDS operand must be: 
  
  \[
  \text{FIELDS}=(5,XXX,BI,A) 
  \]

  Where XXX is KEYLEN + 10.
  It must be coded as in **Figure 36**.
- The FILSZ=En operand is optional. It helps optimize DFSORT main storage and intermediate storage allocation/use. \( n \) is the estimated number of records to be sorted.

**MODS control statement**

 Associates the PSSR program modules with the specific exits (E15 and E35) in the DFSORT program.

The MODS control statement is a required control statement.

If you are using DFSORT Release 8 or later releases, code as shown in [Figure 35 on page 175](#).

**END control statement**

The END control statement causes DFSORT to discontinue reading the SYSIN data set.

The END control statement is a required control statement.

**RECORD control statement**

The RECORD control statement provides record length and type information.

The RECORD control statement is an optional control statement.
- The TYPE=V operand is optional. It indicates that the records to be sorted are EBCDIC variable-length. It must be coded as shown in the following figure.
- Because the records being sorted are variable-length, the optional LENGTH operand can be used to optimize performance. See the DFSORT Application
Programming Guide for more information about how to code the parameters l1, l2, l3, l4, and l5 on the LENGTH operand.

//SYSIN DD *
SORT FIELDS=(5,20,B1,A),FILSZ=E
RECORD TYPE=V,LENGTH=(11,12,13,14,15)
MODS E15=(FABSRT15,50000,IMS2,N),E35=(FABSRT35,50000,IMS2,N)
END
/*

Figure 37. SYSIN data set format (RECORD control statement)

FABSPSSR PSSRIN data set

The PSSRIN control data set contains your specifications for a data compression of *CS format unloaded data set for PSSR.

Subsections:

- “Format”
- “COMPRESSED control statement”

Format

PSSRIN usually resides in the input stream. However, it can be defined either as a sequential data set or as a member of a partitioned data set. It must contain exactly one 80-byte fixed-length record. BLKSIZE, if coded, must be a multiple of 80. The PSSRIN data set can be coded as shown in the following example.

//PSSRIN DD
COMPRESSED=YES
/

Figure 38. PSSRIN data set format

COMPRESSED control statement

Specifies whether the input unloaded records in the *CS format are to be compressed by a segment edit/compression exit or not.

This control statement is optional. If you do not specify this control statement, COMPRESSED=YES is used.

This control statement is effective only when all of the following conditions are met:

- The format of the unloaded database data set is *CS.
- The data compression is specified in the DBD.
- The DBD is defined as non-HALDB.

If the listed conditions do not apply, then this statement is ignored.
YES Specifies that the input unloaded data set is in the *CS format and the data is compressed. PSSR assumes the input unloaded records as compressed.

NO Specifies that the input unloaded data set is the *CS format and the data is not compressed.

**FABSPSSR CTL data set**

The CTL data set contains the control statements which describe the processing to be done by PSSR. It includes the name of the database to be reloaded, along with any optional user requests.

Subsections:
- "Format"
- "Record format"

**Format**

This data set usually resides in the input stream. However, it can be defined either as a sequential data set or as a member of a partitioned data set. It must contain exactly one 80-byte fixed-length record. BLKSIZE, if coded, must be a multiple of 80. The CTL data set can be coded as shown in the following figure.

```
//CTL DD *
PARTSDBD ALL /*
```

*Figure 39. CTL data set format*

**Record format**

There is only one record type in the CTL data set.

```
1---------11-------20-------29---34-----41---46---51-----58--62-65-------------80
dbdname  mod   rbn  anch blk  kstr klen bytes  rng - num
```

**Position**

**Description**

1 This required 8-character field contains the (left-aligned) name of the DBD for the database to be reloaded.

11 This optional 8-character field contains the (left-aligned) name of a user-supplied randomizing module that is used to store and access segments in this HDAM database. It is used to override the (default) DBD value. The randomizing module must have the capability of AMODE=31.
Module FABSRT15 does not establish an IMS environment because it is a sort exit. Consequently, when control is passed to the user-supplied randomizing module, registers 0 and 7 do not point to the IMS control blocks. Each points to a dummy block of 100 bytes, which are initialized to binary zeros. The other interface registers, 1, 9, 13, 14, and 15, are used in PSSR as they are used in IMS.

20 This optional 7-digit field contains the maximum relative block number value that a randomizing module might produce for this database. This value determines the number of control intervals or blocks in the root addressable area of an HDAM database. It must be an unsigned decimal integer. (Use leading zeros, if necessary.) This field is used to override the (default) DBD value.

29 This optional 3-digit field contains the number of RAPs desired in each control interval or block in the root addressable area of an HDAM database. It must be an unsigned decimal integer. (Use leading zeros, if necessary.) This field is used to override the (default) DBD value.

34 This optional 5-digit field contains the database block size (or control interval size). It must be an unsigned decimal integer. (Use leading zeros, if necessary.) This field is used to override the (default) DBD value.

41 This optional 3-digit field lists the starting position of the key field in terms of bytes relative to the beginning of the segment. It must be an unsigned decimal integer. (Use leading zeros, if necessary.) The starting position for the first byte of a fixed-length segment is 001. This field is used to override the (default) DBD value.

46 This optional 3-digit field lists the length of the root segment’s key field in bytes. It must be an unsigned decimal integer. (Use leading zeros, if necessary.) This field is used to override the (default) DBD value.

51 This optional field is used to override the (default) DBD value and prevent cascading.

If the non-zero byte limit value is specified for your database and you are going to reload the database by IMS High Performance Load for z/OS Version 2 Release 1, you must specify ‘NO’ in this field. Otherwise, IMS HP Load V2 ends abnormally with message HPSR3630E. To prevent cascading and avoid degradation of performance in reloading, it is recommended that you use the SORT=YES and the SEARCH=-nnn parameter instead of using PSSR. (For more information about preventing cascading, see “Example 2: Preventing cascading” on page 215.)

The following entries can be made:

5-digit number
The maximum number of bytes of a database record that can be stored into the root addressable area in a series of inserts unbroken by a call to another database record. If you reload a database without using IMS High Performance Load for z/OS Version 2 Release 1, cascading can be prevented when you specify the number of bytes which is less or equal to the available space. The number must be an unsigned decimal integer. (Use leading zeros, if necessary.)

NO There is no limit on the maximum number of bytes of a database record that can be inserted into this database’s root segment addressable area. No attempt to prevent cascading is made. (This entry must be left-aligned.)
This optional 3-digit field lists the range size to be used for the histogram in the Record Size Distribution report. It must be an unsigned decimal integer. (Use leading zeros, if necessary.) The default is 050.

This optional entry can contain any of the following codes:

-   The Record Sizes report or the Record Size Distribution report is not printed.
0   No data is to be sorted. (The run only prints the reports.)
Blank  The Record Size Distribution report is printed.

This optional entry is the maximum number of database record sizes that can be printed in the Record Sizes report. It is used to control the size of the report. The following entries can be made:

number  This number must be a 6-digit unsigned decimal integer. (Use leading zeros, if necessary.)
Blank   The Record Sizes report is not printed.
ALL     All database record sizes are printed on the Record Sizes report.

The parameters at positions 34 and 51 are not effective for HIDAM, HISAM, and SHISAM databases.
PSSR FABSPSSI input

The input data sets differ by the two cataloged procedures. Use the following topics to locate the control statement information for the IMS environment (FABSPSSI procedure).

The following input data sets are used in the IMS environment, typically by the FABSPSSI procedure:

- SORTCNTL data set
- CTL data set
- PSSRIN data set

FABSPSSI SORTCNTL data set

Use this data set only when you want to specify your own values of sort control statement in DFSORT.

The SORTCNTL data set is used only in an IMS environment.

If you do not specify SORTCNTL, PSSR sets default values for DFSORT as follows:

Important: Do not change the values underlined in these examples.

HDAM

SORT FIELDS=(5,20,BI,A),FILSZ=E9400000
MODS E15=(FABSRT15,50000,,N),E35=(FABSPRT35,50000,,N)

HIDAM, HISAM, or SHISAM:

SORT FIELDS=(5,nnn,BI,A),FILSZ=E9400000
MODS E15=(FABSRT15,50000,,N),E35=(FABSPRT35,50000,,N)

PHDAM

SORT FIELDS=(25,2,BI,A,5,20,BI,A),FILSZ=E9400000
MODS E15=(FABPSE15,50000,,N),E35=(FABPSE35,50000,,N)

PHIDAM

SORT FIELDS=(5,nnn,BI,A),FILSZ=E9400000
MODS E15=(FABPSE15,50000,,N),E35=(FABPSE35,50000,,N)

Where nnn is the length of the root key plus 10.

Format

This control data set usually resides in the input stream. It can, however, be defined as a sequential data set or as a member of a partitioned data set. It usually contains 80-byte fixed-length records. Do not specify RECFM=U.

The format is as follows:

```
//SORTCNTL DD *
SORT FIELDS=(25,2,BI,A,5,20,BI,A),FILSZ=En
MODS E15=(FABPSE15,50000,,N),E35=(FABPSE15,50000,,N)
END
/*
```

Chapter 12. Sorting an unloaded database data set with PSSR 181
FABSPSSI PSSRIN data set

The PSSRIN control data set contains your specifications for a user exit, a DATXEXIT and data compression of *CS format for PSSR.

These specifications are effective when PSSR runs in the IMS environment.

Subsections:
- “Format”
- “USEREXIT control statement”
- “DATXEXIT control statement”
- “COMPRESSED control statement” on page 183

Format

PSSRIN usually resides in the input stream. However, it can be defined either as a sequential data set or as a member of a partitioned data set. It must contain exactly one 80-byte fixed-length record. BLKSIZE, if coded, must be a multiple of 80. The PSSRIN data set can be coded as follows:

```
//PSSRIN DD *
USEREXIT=UEXPFGMA,L
DATXEXIT=YES
COMPRESSED=YES
/*
```

USEREXIT control statement

Specifies whether PSSR invokes a user exit routine.

This control statement is optional. If you do not specify this control statement, no user exit routine is invoked.

```
>>>USEREXIT=exitname,L
```

exitname

The name of a user exit routine. Specify this operand if you use the exit interface of PSSR. The specified exit routine must reside in the IMS2 DD data set.

L

Specify this operand only if the user exit routine is to be run under an environment of LE (Language Environment). This option is effective when the user exit routine is written in Enterprise COBOL for z/OS or Enterprise PL/I for z/OS. This option is not effective for Assembler.

DATXEXIT control statement

Specifies whether PSSR invokes a data conversion exit routine.
This control statement is optional. If you do not specify this control statement, PSSR does not activate the data conversion exit.

![DATXEXIT control statement]

YES Activates the data conversion exit.
NO Deactivates the data conversion exit.

The conjunction of the DATXEXIT using of unload utilities of IMS High Performance Unload, PSSR, and IMS HP Load must be one of the following:

Reorganization in the stored form
Specify NO for all three jobs.

Reorganization in the application view form
Specify YES for all three jobs.

**COMPRESSED control statement**

Specifies whether the input unloaded records in the *CS format are to be compressed by a segment edit/compression exit or not.

This control statement is optional. If you do not specify this control statement, COMPRESSED=YES is used.

This control statement is effective only when all of the following conditions are met:

- The format of the unloaded database data set is *CS.
- The data compression is specified in the DBD.
- The DBD is defined as non-HALDB.

If the listed conditions do not apply, then this statement is ignored.

![COMPRESSED control statement]

YES Specifies that the input unloaded data set is in the *CS format and the data is compressed. PSSR assumes the input unloaded records as compressed.
NO Specifies that the input unloaded data set is the *CS format and the data is not compressed.

**FABSPSSI CTL data set**

The CTL data set contains the control statements which describe the processing to be done by PSSR.

Two types of statements are used in the CTL data set: the DB statement and the PART statement.
A DB statement is used for HALDB or non-HALDB. It includes the name of the database to be processed, along with any optional user requests. If HALDB, these user requests are applied to all partitions.

A PART statement is used for HALDB. It includes the name of the partition to be processed, along with any optional user requests. These user requests are applied to the partition, and the user requests by PART statement have priority over the user requests by DB statements.

Subsections:
- “Format”
- “DB statement record format: Line 1”
- “DB statement record format: Line 2” on page 186
- “PART statement record format: Line 1” on page 187
- “PART statement record format: Line 1 and after” on page 188

Format

This data set usually resides in the input stream. However, it can be defined either as a sequential data set or as a member of a partitioned data set. It must contain exactly one 80-byte fixed-length record. BLKSIZE, if coded, must be a multiple of 80.

**DB statement record format: Line 1**

<table>
<thead>
<tr>
<th>Position</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Required. Left-aligned. The name of the DBD for the database to be processed.</td>
</tr>
<tr>
<td>11</td>
<td>Optional. 1- to 8-character, left-aligned. This field contains the name of a user-supplied randomizing module for HDAM or PHDAM database. Use this field to override the (default) DBD value. (^1) The randomizing module must have the capability for AMODE=31.</td>
</tr>
<tr>
<td>20</td>
<td>Optional. Used to override the (default) DBD value. It must contain the maximum relative block number that you want to allow a randomizing module to produce for this database. This value determines the number of control intervals or blocks in the root addressable area of an HDAM database or in each partition of PHDAM. It must be a 7-digit, unsigned decimal integer (with leading zeros, if necessary). (^1)</td>
</tr>
<tr>
<td>29</td>
<td>Optional. Used to override the (default) DBD value. It contains the number of RAPs desired in each control interval or block in the root addressable area of an HDAM database or in each partition of PHDAM. It must be a 3-digit, unsigned decimal integer (with leading zeros, if necessary). (^1)</td>
</tr>
</tbody>
</table>

---

1. The override value applies to all partitions except those otherwise specified by the PART statement.
Optional. Used to override the (default) DBD value. It contains the size of the database block (or control interval) of an HDAM database or in each partition PHDAM. It must be a 5-digit, unsigned decimal integer (with leading zeros, if necessary).

Optional. Used to override the (default) DBD value. This value must be the starting position of the key field in bytes relative to the beginning of the segment. It must be a 3-digit, unsigned decimal integer (with leading zeros, if necessary). The starting position for the first byte of a fixed-length segment is 001.

Optional. Used to override the (default) DBD value. This value must be the length of the root segment’s key field in bytes. It must be a 3-digit, unsigned decimal integer (with leading zeros, if necessary).

Optional. Used to override the (default) DBD value. It contains the byte limit parameter of an HDAM database or in each partition of PHDAM.

If the non-zero byte limit value is specified for your database and you are going to reload the database by IMS High Performance Load for z/OS Version 2 Release 1, you must specify ‘NO’ in this field. Otherwise, IMS HP Load V2 ends abnormally with message HPSR3630E. To prevent cascading and avoid degradation of performance in reloading, it is recommended that you use the SORT=YES and the SEARCH=nnn parameter instead of using PSSR. (For more information about preventing cascading, see “Example 2: Preventing cascading” on page 215.)

It can contain either of the following two kinds of strings:

**5-digit number**

This is the maximum number of bytes of a database record that can be stored in the root addressable area in a series of inserts unbroken by a call to another database record. If you reload a database without using IMS High Performance Load for z/OS Version 2 Release 1, cascading can be prevented when you specify the number of bytes which is less or equal to the available space. It must be a 5-digit, unsigned decimal integer (with leading zeros, if necessary).

**NO**

Means that there is no limit for the maximum number of bytes of a database record that can be inserted into this database’s root segment addressable area. No attempts to prevent cascading is made. (This entry must be left-aligned.)

Optional. This 3-digit field specifies the range of the size to be used for the histogram in the Record Size Distribution report. It must be an unsigned decimal integer (with leading zeros, if necessary). The default is 050.

Optional. Can contain any one of the following codes:

- Requests that the Record Size report and the Record Size Distribution report are not printed.

0 Requests that no data be sorted. The run only prints the reports.

P Requests that the Record Size Distribution report include information about each partition. The partitions that do not include any records are not printed. This option is effective for HALDB.

**Blank** Requests that the Record Size Distribution report is printed. For HALDB, the Record Size Distribution report for each partition is not printed.
Optional. Can contain one of following three codes:

- Requests that the DBD Parameters and Overrides reports are not printed.

P Requests that the DBD Parameters and Overrides reports are printed with all partition override information.

Blank Requests that the DBD Parameters and Overrides reports are printed without the partition information. Blank is the default.

Optional. Can contain one of the following three codes:

- Requests that the HALDB Process Summary report is not printed.

S Requests that the HALDB Process Summary report is printed with all partition information.

Blank Requests that the HALDB Process Summary report is printed with the processed partition information.

Optional. This entry is the maximum number of database record sizes that can be printed in the Record Sizes report. It is used to control the size of the report. It can contain one of three codes:

6-digit number
This number must be a 6-digit unsigned decimal integer (with leading zeros, if necessary).

Blank The Record Sizes report is not printed.

ALL All database record sizes are printed on the Record Sizes report.

Optional. Can contain one of the following three codes:

S Requests that the module name of the partition selection exit be changed, or that the partition selection method be replaced with the partition selection exit. The module name for the new partition selection exit has to be defined on the second line of the DB statement.

H Requests that the method for partition selection be replaced with the high key method.

Blank Requests not to replace the method for partition selection.

**DB statement record format: Line 2**

<table>
<thead>
<tr>
<th>Position</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>Optional. Give the 1-to-8 character left-aligned, name of a user-supplied partition selection exit module. It is used to override the (default) DBD value. If &quot;S&quot; is specified in column 72 of the preceding DB statement, this field is required.</td>
</tr>
</tbody>
</table>
Requirement: If this field is specified, you must put the partition selection exit module in the IMS2 data set of the PSSR JCL, and put the partition selection exit module that is defined in the RECON data set, in the STEPLIB data set of the PSSR JCL.

PART statement record format: Line 1

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dbname</td>
<td>Required. The name of the partition to be processed: 1- to 7-characters, left-aligned.</td>
</tr>
<tr>
<td>mod</td>
<td>Optional. The name of a user-supplied randomizing module for this partition: 1- to 7-characters, left-aligned. You can use it to override the (default) DBD value. This option is effective only for PHDAM.</td>
</tr>
<tr>
<td>rbn</td>
<td>Optional. The maximum relative block number that you want to allow a randomizing module to produce. This value determines the number of control intervals or blocks in the root addressable area of the partition. It must be a 7-digit, unsigned decimal integer (with leading zeros, if necessary). You can use it to override the (default) DBD value. This option is effective only for PHDAM.</td>
</tr>
<tr>
<td>anch</td>
<td>Optional. The number of RAPs desired in each control interval or block in the root addressable area of the partition. It must be a 3-digit, unsigned decimal integer (with leading zeros, if necessary). You can use it to override the (default) DBD value. This option is effective only for PHDAM.</td>
</tr>
<tr>
<td>blksz</td>
<td>Optional. The database block size (or the control interval size). It must be a 5-digit, unsigned decimal integer (with leading zeros, if necessary). You can use it to override the (default) DBD value. This option is effective only for PHDAM.</td>
</tr>
<tr>
<td>fbf</td>
<td>Optional. The free block frequency factor. It must be a 3-digit, unsigned decimal integer (with leading zeros, if necessary). You can use it to override the (default) DBD value. This option is effective only for PHDAM.</td>
</tr>
<tr>
<td>fs</td>
<td>Optional. The free space frequency factor. It must be a 2-digit, unsigned decimal integer (with leading zeros, if necessary). You can use it to override the (default) DBD value. This option is effective only for PHDAM.</td>
</tr>
<tr>
<td>bytes</td>
<td>Optional. The maximum number of bytes of a database record that can be stored into the root addressable area in a series of inserts unbroken by a call to another database record.</td>
</tr>
<tr>
<td>-</td>
<td>If the non-zero byte limit value is specified for your database and you are going to reload the database by IMS High Performance Load for z/OS Version 2 Release 1, you must specify 'NO' in this field. Otherwise, IMS HP Load V2 ends abnormally with message HPSR3630E. To prevent cascading and avoid degradation of performance in reloading, it is recommended that you use the SORT=YES and the SEARCH=nnn parameter instead of using PSSR. (For more information about preventing cascading, see “Example 2: Preventing cascading” on page 215.)</td>
</tr>
</tbody>
</table>

Chapter 12. Sorting an unloaded database data set with PSSR
It can contain either of the following two kinds of strings:

5-digit number
This is the maximum number of bytes of a database record that can be stored in the root addressable area in a series of inserts unbroken by a call to another database record. If you reload a database without using IMS High Performance Load for z/OS Version 2 Release 1, cascading can be prevented when you specify the number of bytes which is less or equal to the available space. It must be a 5-digit, unsigned decimal integer (with leading zeros, if necessary).

NO Means that there is no limit for the maximum number of bytes of a database record that can be inserted into this database’s root segment addressable area. No attempts to prevent cascading is made. (This entry must be left-aligned.)

You can use it to override the (default) DBD value. This option is effective only for PHDAM.

63 Optional. Can contain one of the following three codes:
- Requests that the partition information of DBD Parameters and Overrides reports for this partition are not printed.
Blank Requests that the partition information of DBD Parameters and Overrides reports for this partition are printed. Blank is the default.

72 Optional. Can contain one of the following two codes:
+ The partition selection string or the partition high key is coded on the next line.
Blank The partition selection string and the partition high key is not coded on the next line.

PART statement record format: Line 2 and after

-2--------------------------------------------------------------------------------------------------72-----80
X'xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx '+'
C'ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
+ The partition selection string or the partition high key is continued on the next line.

Blank The partition selection string or the partition high key ends on this line.
PSSR output

PSSR output consists of the unloaded database and some printed reports.

The unloaded database records are contained in the following data sets:

For non-HALDB databases

**SORTOUT data set**
Contains segment records that will be reloaded into the home block.

**SORTOUT2 data set**
Contains segment records whose root segment will not fit into their respective home blocks if the input DBD organization is HDAM.

These data sets are then used as input to either the IMS HD Reorganization Reload utility or its equivalent reload utility. When used in the reload step, they must be concatenated, with SORTOUT being the first data set.

**Important:** If you reload an HDAM database by using IMS High Performance Load for z/OS Version 2 Release 1 or later, you cannot use SORTOUT2 as an input for the reload. For details, see "Considerations before using PSSR" on page 155.

For HALDB databases

Either of the following data sets:

**SORTOUT data set**
Contains segment records sorted in partition sequence and, in each partition, sorted in physical sequence. Any segment record whose root segment does not fit into its home block (according to the PSSR algorithm) are placed at the end in each partition.

**Output partition data set**
Contains all the segment records that need to be reloaded into one partition.

Unlike non-HALDB databases, the SORTOUT2 data set is used as a work data set for PSSR. You must not reload the segments that are contained in this data set.

If the randomizing routine returns a return code of 4 for a root segment, the database record that contains that root segment is put into the SORTOUT3 data set, if it is specified in the JCL. This data set must not be used as an input for the previously mentioned reload utilities.

If the DL/I partition selection returns a nonzero return code for HALDB processing, and if SORTOUT4 is specified in the JCL, the database record that contains the root segment is put into the SORTOUT4 data set. This data set must not be used as an input for the previously mentioned reload utilities.

If the SORTOUTD DD is specified, the database record that contains the duplicate root key segment is put into the SORTOUTD data set. You must determine whether this data set is to be used or not as an input for the previously mentioned reload utilities.

The PSSR reports are generated in the following five data sets:

- SYSOUT: produced by DFSORT
SORTOUT data set for non-HALDB processing

The SORTOUT data set is the primary PSSR output data set. It is the sorted output from the DFSORT program. This data set contains most or all of the database segments, sorted in the physical (hierarchical) sequence for HDAM, or sorted into root key sequence for HIDAM, HISAM, or SHISAM.

Format

The SORTOUT data set, when concatenated with the SORTOUT2 data set, is in exactly the same format as the unloaded database produced by the Unload utility. If the record length is changed by a user exit routine, the format of the SORTOUT data set needs to be changed equally. It contains a table record, followed by records that contain database segments.

PSSR uses the bytes parameter (from the control statement of the CTL data set or from the DBD) to determine which each root segment is likely to be placed in its home block (that is, the block to which that root segment is assigned by the randomizing routine) for HDAM processing. Every database record whose root segment will fit into its home block (according to the PSSR algorithm) is placed in the SORTOUT data set. All database records are placed in the SORTOUT data set for HIDAM, HISAM, or SHISAM processing.

SORTOUT is a sequential data set that can be on a tape or disk.

SORTOUT2 data set for non-HALDB processing

The SORTOUT2 data set is produced by module FABSRT35. It contains all of the database segments that are not written to the SORTOUT data set.

The database records put into SORTOUT2 are not in the SORTOUT data set. If the database is a HIDAM, a HISAM, or a SHISAM database, no data is contained.

Important: If you reload an HDAM database by using IMS High Performance Load for z/OS Version 2 Release 1 or later, the SORTOUT2 data set cannot be used as an input for reload. For the details, see “Considerations before using PSSR” on page 155.

Format

The SORTOUT2 data set, when concatenated after the SORTOUT data set, is in exactly the same format as the unloaded database produced by the IMS HD Reorganization Unload utility. It contains records containing database segments, followed by a table record.

PSSR uses the bytes parameter (from the control statement of the CTL data set or from the DBD) to determine if each root segment is likely to be placed in its home block (that is, the block to which that root segment is assigned by the randomizing routine). All database records of which root segment will not fit into its home block (according to the PSSR algorithm) are placed in the SORTOUT2 data set.
SORTOUT2 is a sequential data set that can be on tape or disk.

**SORTOUT data set for HALDB processing**

This data set is a required output data set. Segment records that should be reloaded and not put into an output partition data set are all placed in this data set.

The records in this data set are sorted in physical sequence within each partition; the partitions are sorted in the order in which they are selected. The segment records whose root segment does not fit into home block (according to the PSSR algorithm) are placed at the end of each partition.

**Format**

The SORTOUT data set is a sequential data set. Basically, it has exactly the same format as the unloaded database data set that is produced by the Unload utility. If the record length is changed by a user exit routine, the format of the SORTOUT data set needs to be changed equally. The following figure shows an example of how the segment records in a partition data set are arranged:

```
<table>
<thead>
<tr>
<th>Header record for the data set</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
</tr>
<tr>
<td>Physically sorted segment records (in partition 1) that will be reloaded into the home block</td>
</tr>
<tr>
<td>...</td>
</tr>
<tr>
<td>...</td>
</tr>
<tr>
<td>Physically sorted segment records (in partition 1) whose root segments will not fit into the home block</td>
</tr>
<tr>
<td>...</td>
</tr>
<tr>
<td>...</td>
</tr>
<tr>
<td>Physically sorted segment records (in partition 2) that will be reloaded into the home block</td>
</tr>
<tr>
<td>...</td>
</tr>
<tr>
<td>...</td>
</tr>
<tr>
<td>Physically sorted segment records (in partition 2) whose root segments will not fit into the home block</td>
</tr>
<tr>
<td>...</td>
</tr>
<tr>
<td>...</td>
</tr>
<tr>
<td>Trailer record for the data set</td>
</tr>
</tbody>
</table>
```

*Figure 40. Example of the SORTOUT data set*

**Note:** If the DD statements for all output partition data sets are specified in your PSSR JCL, no segment record is put into the SORTOUT data set.

**SORTOUT2 data set for HALDB processing**

This data set is a required work data set for PSSR.

You must not reload the segments that are contained in this data set for HALDB.

**Format**

The data set has the same format as SORTOUT.
**SORTOUT3 data set**

This data set is an optional output data set. This data set, if specified, contains the database records whose root segment key received a return code of 4 from the randomizing routine.

If the DD statement for this data set is not specified in the JCL or is coded as DUMMY, such database records are discarded.

The database records put into SORTOUT3 are not contained in the SORTOUT data set, the Output partition data set, the SORTOUT2 data set, or in the SORTOUT4 data set.

**Format**

The data set has the same format as SORTOUT.

**SORTOUT4 data set**

This data set is an optional output data set. This data set, if specified, contains the database records whose root segment keys are returned with a nonzero return code by the partition selection.

If the DD statement for this data set is not specified in the JCL, or is coded as DUMMY, such database records are discarded.

The database records put into SORTOUT4 are not contained in the SORTOUT data set, the Output partition data set, the SORTOUT2 data set, or in the SORTOUT3 data set.

**Format**

The data set has the same format as SORTOUT.

**SORTOUTD data set**

This data set is an optional output data set. If it is specified, PSSR checks for a duplicate root key.

When SORTOUTD is specified, and the root key is defined as unique and root keys is not defined as compressed in the DBD, PSSR checks every value of the root key. If some root keys have the same value, only the first database record input to PSSR is stored into the SORTOUT, SORTOUT2, or output partition data set. The other root keys, which are named as duplicated records, are stored into this SORTOUTD data set. If the SORTOUTD is coded as DUMMY, such duplicated records are discarded.

If the database records are changed by a user exit, the key duplications are checked with the changed root keys, and the changed database records are contained in the SORTOUTD data set.

**Format**

The data set format has the same as SORTOUT.
Output partition data set

This output data set is unique to HALDBs. It is a sequential data set that contains all the segment records that should be reloaded into a partition. The name of the DD is the partition name that is defined in the RECON data sets.

Format

Although the records contained in each partition data set are the segments to be reloaded into one partition, each output partition data set has the same format as the SORTOUT data set.

By specifying the names of all the partitions in your HALDB, you can split your unloaded database data set into a set of partition data sets. You can reload these partition data sets using IMS HP Load. The following figure shows an example of how the segment records in a partition data set are arranged:

<table>
<thead>
<tr>
<th>Header record for the partition data set of HALDB</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
</tr>
<tr>
<td>Physically sorted segment records for the partition that will be reloaded into the home block</td>
</tr>
<tr>
<td>...</td>
</tr>
<tr>
<td>Physically sorted segment records for the partition whose root segments will not fit into the home block</td>
</tr>
<tr>
<td>...</td>
</tr>
<tr>
<td>Trailer record for the data set</td>
</tr>
</tbody>
</table>

*Figure 41. Example of the output partition data set*

SYSOUT data set

This data set is the system output data set for messages from the DFSORT program.

Format

Code your DD statement as follows:

```
//SYSOUT DD SYSOUT=A
```

SYSPRINT data set

This data set is the system output data set for any messages from IMS.

Format

Code your DD statement as follows:

```
//SYSPRINT DD SYSOUT=A
```

PR8 data set

This data set contains the reports that are produced by module FABSRT15 or FABSPE15.

The following reports are contained in this data set:
- Control Card Format report
- Control Card report
- DBD Parameters and Overrides report
- Record Sizes report
- Comments on DL/I HDAM Database Load Procedure report
- Record Size Distribution report

**Format**

This data set contains 133-byte fixed-length records, and block size (if coded in your JCL) must be a multiple of 133. You must code your DD statement as follows:

```plaintext
//PRB DD SYSOUT=A
```

**Control Card Format report**

This report describes the fields on the records in the CTL data set. It is printed for the convenience of the user.

The following figure shows an example of this report.

---

**Figure 42. Control Card Format report**

**Control Card report**

This report contains a printed copy of the control statement that the user provided in the CTL data set.

---

Chapter 12. Sorting an unloaded database data set with PSSR 195
The following figure shows an example of this report.

**DBD Parameters and Overrides report**

This report contains a printed copy of the DBD parameters that were used in the PSSR job. Any parameters that were overridden by the user’s control statement are flagged ("**" is printed next to the overridden value).

The following figure shows an example of this report.

**Record Sizes report**

This report contains a printed list of the sizes of all database records.

If the PAD operand of COMPRTN= parameter in DBDGEN SEGM statement is defined, and the length of the fixed-length segment is shorter than the PAD value by the Segment Edit/Compression exit, the PAD value is used to calculate the size of the database records instead of the segment length.

This report also contains the total number of root segments and the total number of dependent segments that were processed. Ten record sizes are printed on each line. Therefore, the number of printed lines on this report is about one tenth of the number of root segments in your database.

**Tip:** If your database is large, this report will also be fairly large. You might want to suppress this report by using a blank in column 65 of your control statement (see “FABSPSSR CTL data set” on page 178 or “FABSPSSI CTL data set” on page 183).

The following figure shows an example of this report.
**Figure 45. Record Sizes report**

**Comments on DL/I HDAM Database Load Procedure report**

This report contains some comments that helps you understand **cascading** and how HDAM and PHDAM databases are loaded. It is printed for the convenience of the user.

The following figure shows an example of this report.

**Figure 46. Comments on DL/I HDAM Database Load Procedure report**

**Record Size Distribution report**

This report contains a printed histogram of the sizes of all database records.

If the PAD operand of COMPRTN= parameter in DBDGEN SEGM statement is defined, and the length of the fixed-length segment is shorter than the PAD value by the Segment Edit/Compression exit, the PAD value is used to calculate the size of the database records instead of the segment length.
You can specify the width of the histogram classes in column 58 of your control statement (see “FABSPSSR CTL data set” on page 178 or “FABSPSSI CTL data set” on page 183). For databases with a large number of database records, this report is much more useful than the Record Sizes report.

The following figure shows an example of this report.

```
Figure 47. Record Size Distribution report
```

**PR9 data set**

This data set contains the Assigned Roots per RAP report, which is produced by module FABSRT35 or FABSPES35.

**Format**

This data set contains 133-byte fixed-length records, and the block size (if coded in your JCL) must be a multiple of 133. You must code your DD statement as follows:

```
//PR9 DD SYSOUT=A
```

**Assigned Roots per RAP report**

This report contains a map of all RAPs in the database. It gives the actual number of root segments that randomized to each RAP. It also includes some totals, including the number of database records that did not fit into their home blocks.

The following figure shows an example of this report.
**PR9X data set**

This data set contains the Assigned Roots per Block report, which is produced by module FABSRT35 or FABSPE35.

**Format**

This data set contains 133-byte fixed-length records, and block size (if coded in your JCL) must be a multiple of 133. Code your DD statement as follows:

```
//PR9X DD SYSOUT=A
```

**Assigned Roots per Block report**

This report contains a map of all blocks (or control intervals) in the database. It gives the actual number of root segments that randomized to each block. It also includes some totals.

The following figure shows an example of this report.

---

**Figure 48. Assigned Roots per RAP report**

---
**PR10 data set**

This data set contains the HALDB Process Summary report.

**Format**

This data set contains a 133-byte-fixed-length record.

**HALDB Process Summary report**

This report contains a summary of the processing. The report is printed only for HALDB.

The following figure shows an example of this report.

---

**Figure 50. HALDB Process Summary report**

- **Note:** The partitions that are indicated with asterisks (*) in this report include the segment records after sorting by PSSR job.
Chapter 13. PSSR user exit

PSSR supports user exit routines for deleting and modifying segments.

Programming interface of PSSR user exit

These topics contain product-sensitive programming interface information for PSSR. See the Programming interface information section at the end of this publication to understand the restrictions that are associated with this type of material.

The user exit routine is invoked after every reading of segment data from the HD unloaded data set. You can use the user exit routine to check, modify, or delete the segments.

The user exit module is loaded dynamically by PSSR.

When L is specified in the PSSRIN data set, PSSR establishes the LE environment by using the preinitialization services. The user exit is invoked by the CEEPIPI call_sub function.

Sample user exits

The following sample user exits are available in the HPS.SHPSSAMP data set:

HPSRXAA0:
A sample user exit that is written in the Assembler language.

HPSRXCC1:
A sample user exit that is written in Enterprise COBOL for z/OS.

HPSRXPP1:
A sample user exit that is written in Enterprise PL/I for z/OS.

The following topics describe the user exits of PSSR.

Topics:

- “Program functions of PSSR user exit” on page 202
- “Parameters for PSSR user exit” on page 204
- “Interface of Assembler user exit routine” on page 207
- “Interface of Enterprise COBOL for z/OS program user exit routine” on page 208
- “Interface of Enterprise PL/I for z/OS user exit routine” on page 209
Program functions of PSSR user exit

PSSR supports a user exit routine for deleting and modifying segments. The user exit is invoked by PSSR after every reading of a segment record from the unloaded data set.

If the unloaded database data set is HD, SH or CS format and the DECN option is specified in IMS High Performance Unload for z/OS unload job, PSSR processes as follows:
1. Calls the segment edit/compression exit routine to decompress the segment data before calling the user exit routine.
2. Calls the user exit.
3. Calls the partition selection for a HALDB, and the HDAM randomizing routine.
4. Calls the segment edit/compression exit routine again to recompress the segment data.

If the unloaded database data set is of *CS format and COMPRESSED=YES is specified in PSSRIN, PSSR also processes the same way.

The library that contains the user exit routine must be concatenated to the IMS2 DD statement.

The user exit routine can be coded in Assembler, Enterprise COBOL for z/OS programs, or Enterprise PL/I for z/OS. When the user exit is written in COBOL or PL/I, it is run in the Language Environment (LE).

Restriction: The user exit is supported only when PSSR is running on IMS.

PSSRIN data set

To activate the user exit, specify USEREXIT=modname in the PSSRIN control statement of PSSR job.

In the basic environment (not the LE environment), PSSR supports user exit routines that are written in the Assembler language. In this case, do not specify the second parameter of the USEREXIT statement in the PSSRIN data set of your PSSR JCL. See Programming interface of PSSR user exit on page 201.

In the LE environment, PSSR supports user exit routines that are written in Enterprise COBOL for z/OS or Enterprise PL/I for z/OS. For user exit routines that are written in Enterprise COBOL for z/OS or Enterprise PL/I for z/OS, specify L as the second parameter of the USEREXIT statement in the PSSRIN data set of your PSSR JCL. There are some coding rules for compiling options that you must follow. These rules are documented in Programming interface of PSSR user exit on page 201.

User Exit Action report

A User Exit Action report is added to the Record Sizes report in PR8 data set.

This report includes the following information:

INPUT:

The number of segment occurrences in the unload file. The bottom number
of INPUT indicates the total number of segments in the unload file. The number of INPUT is not equal to the frequency of the user exit called.

**OUTPUT:**
The number of segment occurrences in the output files. The bottom number of OUTPUT indicates the total number of segments in the output files. For non-HALDB, the output files are SORTOUT, SORTOUT2, SORTOUT3, and SORTOUTD. For HALDB, the output files are SORTOUT, output partition data set, SORTOUT3, SORTOUT4, and SORTOUTD.

**DELETED:**
The number of segment occurrences that are deleted by the user exit routine.
Parameters for PSSR user exit

The following information describes the parameters for the user exit and register contents upon return from the user exit.

This topic contains product-sensitive programming interface information for PSSR. See the Programming interface information section at the end of this publication to understand the restrictions that are associated with this type of material.

Subsections:
- “Parameters for the user exit”
- “Register contents upon return from the user exit” on page 205

Parameters for the user exit

The following table contains a list of parameters.

Table 24. Parameters for the user exit

<table>
<thead>
<tr>
<th></th>
<th>Assembler</th>
<th>COBOL</th>
<th>PL/I</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A(segment name)</td>
<td>segment name</td>
<td>pointer of segment name</td>
<td>A 9-byte area that contains an 8-byte segment name and a 1-byte segment code. At termination, the first 8 bytes of the area contain blanks.</td>
</tr>
<tr>
<td>2</td>
<td>A(segment data)</td>
<td>segment data</td>
<td>pointer of segment data</td>
<td>The segment data that is read from the HD unloaded data set. PSSR always decompress compressed data before presenting it to the user exit routine.</td>
</tr>
</tbody>
</table>
Table 24. Parameters for the user exit (continued)

<table>
<thead>
<tr>
<th></th>
<th>Assembler</th>
<th>COBOL</th>
<th>PL/I</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>A(work area)</td>
<td>work area</td>
<td>pointer of work area</td>
<td>A segment work area large enough to hold the largest segment that can be defined in the DBD being used. This area must be used whenever the length of a segment has to be changed. For variable-length segments, you must use the user exit routine to change the segment length (specified in the LL field) to a value that is appropriate for the segment being modified and that does not exceed the maximum segment length defined in the DBD being used. For fixed-length segments, the amount of segment data that is moved into the database data set is controlled by the setting in the DBD. If that setting has been expanded, it is your responsibility to initialize the expanded part of the segment by use of the user exit routine. If the routine issues a return code of zero, PSSR moves unchanged segment data into the database data set, and automatically initializes the expanded part of the segment to binary zeros.</td>
</tr>
<tr>
<td>4</td>
<td>A(database name)</td>
<td>database name</td>
<td>pointer of database name</td>
<td>The 8-byte name of the database that is processed.</td>
</tr>
</tbody>
</table>

- For Assembler language, the address of the parameter list is pointed by register 1.
- For COBOL, each parameters are passed as the real ones, not as pointers.
- For PL/I, you need to declare each parameters as pointers.

Register contents upon return from the user exit

When PSSR returns from the user exit routine, it returns the return code to PSSR. This is necessary so that PSSR can determine the correct process of the segment data. The return code must be one of the values that are listed in the following table.
Table 25. Register contents upon return from the user exit

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Do not modify the segment. PSSR uses the segment data that is pointed to by the second word of the parameter list.</td>
</tr>
<tr>
<td>4</td>
<td>Modify the segment. PSSR uses the segment data that is pointed to by the third word of the parameter list.</td>
</tr>
<tr>
<td>8</td>
<td>Delete the segment. PSSR skips this segment. The user exit routine must return the return code 8 for all dependent segments of this segment.</td>
</tr>
<tr>
<td>12</td>
<td>Delete the segment and all subsequent segments. PSSR skips this segment and all subsequent segments in the same database record.</td>
</tr>
<tr>
<td>100—999</td>
<td>Indicates termination. PSSR ends the process and passes this return code back as the job return code. The value of the return code can be from 100 to 999.</td>
</tr>
</tbody>
</table>

**Note:** A return code of 16 or 20 is interpreted as a return code of 0.
Interface of Assembler user exit routine

When you use an Assembler user exit routine, ensure that the exit routine meets the following requirements.

This topic contains product-sensitive programming interface information for PSSR. See the Programming interface information section at the end of this publication to understand the restrictions that are associated with this type of material.

• It must not be specified L after the user exit name in the PSSRIN data set.
• The Assembler user exit routine is simply called by PSSR. The Language Environment is not established.
• When the PSSR enters the user exit routine, the contents of the registers are as listed in the following table.

Table 26. Interface of Assembler user exit routine

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Parameter list pointer</td>
</tr>
<tr>
<td>13</td>
<td>Register save area pointer</td>
</tr>
<tr>
<td>14</td>
<td>Return address</td>
</tr>
<tr>
<td>15</td>
<td>Entry point address of the user exit</td>
</tr>
</tbody>
</table>

• Before returning to the PSSR, the user exit routine must set the return code in register 15.

A sample of this user exit is HPSRXAA0 in the HPS.SHPSSAMP data set.
Interface of Enterprise COBOL for z/OS program user exit routine

When you use an Enterprise COBOL for z/OS program user exit routine, ensure that the exit routine meets the following requirements.

This topic contains product-sensitive programming interface information. See the Programming interface information section at the end of this publication to understand the restrictions that are associated with this type of material.

• It must be specified L after the user exit name in the PSSRIN data set.
• The user exit routine is called in the Language Environment (LE). LE libraries need to be used as follows:
  – SCEERUN is needed during compiling the user exit and running the PSSR.
  – SCEELKED is needed during the linkage-edit process of the user exit.
• The return code must be set in RETURN-CODE.
• STOP RUN cannot be used in the program.

A sample of this user exit is HPSRXCC1 in the HPS.SHPSSAMP data set.
Interface of Enterprise PL/I for z/OS user exit routine

When you use an Enterprise PL/I for z/OS user exit routine, ensure that the exit routine meets the following requirements.

This topic contains product-sensitive programming interface information. See the Programming interface information section at the end of this publication to understand the restrictions that are associated with this type of material.

- It must be specified L after the user exit name in the PSSRIN data set.
- The user exit routine is called in the Language Environment (LE). LE libraries need to be used as follows:
  - SCEERUN is needed during compiling the user exit and running the PSSR.
  - SCEELKED is needed during the linkage-edit process of the user exit.
- The PL/I user exit routine is loaded dynamically and called by PSSR as a subprogram. It needs to be specified as OPTIONS(FETCHABLE) on the procedure statement in the program. OPTIONS(MAIN) cannot be specified.
- The parameters in the PL/I routine must be declared as pointer variables.
- The PL/I built-in function PLIRETC or LE callable service CEE3SRC can be used to set the return code.
- POSIX(ON) cannot be specified for the LE runtime option.
- Nested or multiple enclaves are not supported. For more information about the nested enclave, see the topic “Using Nested Enclaves” in the Language Environment Programming Guide.

A sample of this user exit is HPSRXPP1 in the HPS.SHPSSAMP data set.
Chapter 14. PSSR usage examples

There are many ways to use PSSR. The examples that are given here represent some of the typical tasks that PSSR can perform. By studying and understanding these examples, you can learn the techniques to use and to reorganize your own databases with PSSR.

Topics:

- “Example 1: Normal reorganization (Changing root addressable area)” on page 212
- “Example 2: Preventing cascading” on page 215
- “Example 3: Iterative tuning analysis” on page 219
- “Example 4: Print reports about record sizes” on page 222
- “Example 5: HALDB processing” on page 224
Example 1: Normal reorganization (Changing root addressable area)

This example shows a scenario in which a single HDAM database is reorganized.

In this example, a single HDAM database is being reorganized. Because of recent growth in the amount of data in the database, the size of the root addressable area is being increased. Such a change completely alters the resulting physical sequence of the root segments. The database segments will be unloaded in the old physical sequence and reloaded in the new sequence. A likely result is excessive disk arm movement during the reload step. This is the kind of situation in which a significant performance improvement is likely to be gained by using PSSR. This example illustrates how to use PSSR in a very common type of database reorganization process.

Because there are no logical relationships in this database, there is no need to execute any of the IMS utilities that handle logical relationships. The following steps are required:

1. “Create an image copy of the old database”
2. “Unload the database with the IMS HD Reorganization Unload utility”
3. “Create a new DBD” on page 213
4. “Sort the unloaded database data set with PSSR” on page 213
5. “Delete the old database cluster and define a new one” on page 213
6. “Reload the database with the IMS HD Reorganization Reload utility” on page 213
7. “Create an image copy of the new database” on page 214
8. “Run the HD Pointer Checker utility with image copy input” on page 214

Create an image copy of the old database

An image copy is created as a protective measure. In case something goes wrong, the database can be recovered with no data loss.

Unload the database with the IMS HD Reorganization Unload utility

The database is unloaded in this step. It is a standard database unload execution.

Tip: This step can be replaced with an equivalent program, such as the unload utilities of IMS High Performance Unload.

```
//IMSHDUN EXEC PGM=DFSRRC00,
//       PARM='ULU,DFSURGU0,ORDHDM,...........,N,N'
//IMS DD DSNAME=HPS.TEST.DBDLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A
//DFSURGU1 DD DSNAME=UNLOAD.FILE,DISP=(NEW,PASS),
//           UNIT=SYSDA,SPACE=(CYL,(1,1))
//DFSRESLB DD DSNAME=IMSVS.RESLIB,DISP=SHR
//DFSVSAMP DD DSNAME=HPS.TEST.SOURCE(DFSVSAMP),DISP=SHR
//EDS1HDM1 DD DSNAME=HPS.EDS1HDM1,DISP=OLD
```

Figure 51. Example 1: IMS HD Reorganization Unload (DFSURGU0) JCL
Create a new DBD

In this example, the DBD is modified to increase the size of the root addressable area. This step might be required because the amount of data in the database increased and caused performance degradation. This type of DBD change does not require a new PSB.

Sort the unloaded database data set with PSSR

The unloaded database is sorted in this step in order to make the reload step run faster. The cost of running this step is expected to be offset by a larger reduction in the cost of running the reload step.

The control statement of the CTL data set is in the standard format used for reloading a large database. Because column 65 is blank, the potentially gigantic Record Sizes report is not printed. All other PSSR reports are generated.

```
//DFSORT EXEC FABSPSSR,
// DDBLIB='IMSVS.DBDLIB'
//STEPHD.CTL DD *
ORDHDM *
//STEPHD.SYSIN DD *
  SORT FIELDS=(5,20,BI,A),FILSZ=E9400000
  MODS E15=(FABSRT15,50000,IMS2,N),E35=(FABSRT35,50000,IMS2,N)
END /
```

Figure 52. Example 1: DFSORT (PSSR) JCL

Delete the old database cluster and define a new one

Because the database is VSAM, a new cluster must be defined.

Reload the database with the IMS HD Reorganization Reload utility

The database is reloaded in this step. Except for the DFSUINPT DD statement, it is a standard database reload execution.

The DFSUINPT data set contains the data to be reloaded. It is normally the data set created by the IMS HD Reorganization Unload utility. In this example, it is a concatenation of the two PSSR output data sets.

Tip: This step can be replaced with an equivalent program, such as the IMS HP Load utility.
Create an image copy of the new database

This image copy will be the first backup of your reorganized database.

Run the HD Pointer Checker utility with image copy input

Run the HD Pointer Checker utility of IMS High Performance Pointer Checker for the following reasons:

- To verify that your reorganization procedure worked properly
- To verify that your image copy is usable
- To obtain statistical reports that describe your database when it is in its optimally reorganized condition

Figure 53. Example 1: IMS HD Reorganization Reload (DFSURGLO) JCL
Example 2: Preventing cascading

Cascading can be prevented or reduced by changing certain DBD parameters or by using PSSR.

Important: If you reload a database by using IMS High Performance Load for z/OS Version 2 Release 1, you cannot use PSSR to prevent cascading. To prevent cascading and avoid degradation of performance in reloading, it is recommended that you use the SORT=YES and the SEARCH=nnn parameter instead of using PSSR.

Subsections:
- "Changing DBD parameters"
- "Using PSSR"
- “Example of steps for preventing cascading” on page 217
- “Sort the unloaded database data set with PSSR” on page 217

Changing DBD parameters

The following DBD parameters can affect the tendency of a database to exhibit cascading during a reload in physical sequence:
- RMNAME keyword on the DBD statement
- BLOCK keyword on the DATASET statement
- SIZE keyword on the DATASET statement
- SCAN keyword on the DATASET statement
- FRSPC keyword on the DATASET statement

By tuning the DBD to better match the data, you might be able to reduce or eliminate cascading. You can use the reports from PSSR.

Using PSSR

PSSR can also prevent cascading during reorganization by altering the reload sequence. This alteration causes some database records to be loaded at the end of the reload, instead of in their normal physical sequence. Those records will not contribute to the cascading effect. If the few troublesome records are held back until the other records are loaded, the cascading problem may be avoided.

As it processes the records that have randomized to a particular block, PSSR keeps track of the number of unused bytes in the block. Once this number gets smaller than the length of a root segment, PSSR sets an internal flag. All subsequent database records for that block are considered to be the troublesome ones.

The output unloaded database is contained on two sequential data sets:
- The SORTOUT data set contains all of the database records that are expected to fit comfortably in their home blocks. The segments that make up these records are sorted into their physical hierarchical sequence. Disk arm movement is minimized when reloading from this data set. Most of the database records are usually contained in the SORTOUT data set.
- The SORTOUT2 data set contains the rest of the database records. These troublesome records are expected to spill out of their home blocks if they are loaded in their natural physical sequence.
When you run the IMS HD Reorganization Reload utility, use both of these data sets as input. The DFSUINPT data set must be a concatenation of both the SORTOUT and the SORTOUT2 data sets.

**Note:** The SORTOUT data set must be the first in the concatenation (see Figure 54). Otherwise, the reload does not work properly.

```plaintext
//DFSUINPT DD DSNAMES=&amp;UNLSRT, SORTOUT data set
//   DISP=(OLD,DELETE,DELETE)
// DFSUINPT DD DSNAMES=&amp;UNLSRT2, SORTOUT2 data set
//   DISP=(OLD,DELETE,DELETE)
```

**Figure 54. The DFSUINPT data set**

In order for this cascading prevention algorithm to be used, you *must* specify a byte limit for your database. You can do this by using one of the following methods:

- In the DBD, use the "BYTES" operand of the "RMNAME" keyword on the "DBD" statement.
- On the PSSR control statement, use the value in column 51.
- Use both the DBD and the PSSR control statements. (The PSSR control statement overrides the DBD.)

If you did not specify the byte limit, or if the byte limit is larger than the available space in one block (see Figure 55), then the cascading prevention algorithm is disabled. All database records are written to the SORTOUT data set.

**Figure 55. Available space in one block**

\[
A = B - 4 - 4R - V - (F*B)/100
\]

where

- \(A\) = available space in a root-addressable-area block
- \(B\) = block size of data set group number 1
- \(R\) = number of RAPS in one block
- \(V\) = \(7 \times (4 \times (CIDF) + 3 \times (RDF))\) if the access method is VSAM
  \(0\) if the access method is OSAM
- \(F\) = free space percentage factor

**Note:** Using the PSSR DBD override does not actually change the DBD. It merely changes the way PSSR sequences the database records.

**Important:** If you reload a database by using IMS High Performance Load for z/OS Version 2 Release 1, you cannot use PSSR to prevent cascading. To prevent cascading and avoid degradation of performance in reloading, it is recommended that you use the SORT=YES and the SEARCH=nnn parameter instead of using PSSR. When the non-zero byte limit value is specified for your database, you must specify 'NO' in columns 51-52 of your control statement. Otherwise, IMS HP Load V2 ends abnormally with message HPSR3630E. For more information, see "FABSPSSR CTL data set" on page 178 or "FABSPSSI CTL data set" on page 183.
Example of steps for preventing cascading

In this example, a single HDAM database is being reorganized. Because the DBD is not being changed during the reorganization, the database segments will be unloaded and reloaded in physical sequence. The particular data in this database makes it especially prone to cascading during a reload performed in physical sequence. Therefore, PSSR is used to alter the reload sequence in an attempt to minimize cascading.

Because there are no logical relationships in this database, there is no need to execute any of the IMS utilities that handle logical relationships. The following steps are required:

1. Create an image copy of the old database.
2. Unload the database with IMS HD Reorganization Unload utility or an equivalent program, such as IMS High Performance Unload for z/OS unload utilities.
3. Delete the old database cluster and define a new one.
4. Reload the database with IMS HP Load V2 using the SORT=YES and the SEARCH=\-nnn parameter.
5. Create an image copy of the new database.
6. Run the IMS High Performance Pointer Checker for z/OS HD Pointer Checker utility with image copy input.

Except for step 4, these steps are identical to the corresponding steps of “Example 1: Normal reorganization (Changing root addressable area)” on page 212. Explanations and JCL for those steps are not repeated in this topic.

Sort the unloaded database data set with PSSR

The unloaded database is sorted in this step (see Figure 56 on page 218) in order to minimize cascading during the reload step. In this example, the cost of running the reload step is not expected to be reduced by running PSSR. The expected benefit from running PSSR is improved performance during production use of the database (because the amount of cascading is reduced).

In the following example, the PSSR control statement causes PSSR to override the DBD. The value in column 51 overrides the BYTES operand of the RMNAME keyword on the DBD statement. Overriding this parameter causes PSSR to place some database records in the SORTOUT2 data set. These records are reloaded last in order to eliminate their contribution to the cascading problem. The value of the override is the maximum amount of space (in one block) that can be used to contain database segments. The PSSR control statement is in the standard format used for reloading a large database in order to minimize cascading. Because column 65 is blank, the potentially gigantic Record Sizes report is not printed. All other PSSR reports are generated.
Important: If you reload a database by using IMS High Performance Load for z/OS Version 2 Release 1, you cannot use PSSR to prevent cascading. To prevent cascading and avoid degradation of performance in reloading, it is recommended that you use the SORT=YES and the SEARCH=-nnn parameter instead of using PSSR. When the non-zero byte limit value is specified for your database, you must specify 'NO' in columns 51-52 of your control statement. Otherwise, IMS HP Load V2 ends abnormally with message HPSR3630E. For more information, see “FABSPSSR CTL data set” on page 178 or “FABSPSSI CTL data set” on page 183.

Figure 56. Example 2: DFSORT (PSSR) JCL

```
//DFSORT EXEC FABSPSSR,
//  DBDLIB='HPS.TEST.DBDLIB'
//STEPHD.CTL DD *
ORDHDM 02028
/*
//STEPHD.SYSIN DD *
  SORT FIELDS=(5,20,BI,A),FILSZ=E9400000
  MODS  E15=(FABSRT15,50000,IMS2),E35=(FABSRT35,50000,IMS2)
END
/*
```
Example 3: Iterative tuning analysis

In this example, PSSR is used to investigate how potential DBD changes affect randomizer performance.

This kind of analysis is usually done in the following situation:
- Some performance degradation is being experienced.
- A database reorganization does not eliminate the poor performance.
- Poor randomization is the suspected cause of the bad performance.

PSSR can model the randomization process without actually creating a reorganized database. It prints the following two reports:
- Assigned Roots per Block, which tells how many root segments were randomized to each database block.
- Assigned Roots per RAP, which tells how many root segments were randomized to each RAP.

You can find out how many root segments are randomized to each block or RAP from these reports.

Note: Although they are much more compact than the Record Sizes report, these reports can still be quite large. They each contain approximately 5000 - 6000 entries on each page. A large database could result in a very large printed report. For example, assume that your database root addressable area is 250,000 blocks long, with two RAPs in each block. The Assigned Roots per Block report will be nearly 500 pages long, and the Assigned Roots per RAP report will be nearly 100 pages long.

Having well-distributed root segments is required to obtain optimum performance from an HDAM database. You want to minimize the number of physical input/output requests needed to access a segment. Having each root segment stored in its home block is a major factor. If your root segments are fairly evenly distributed throughout the root addressable area, it should minimize the number of roots that are not stored in their home block.

Database characteristics usually change over time. This can sometimes contribute to degraded performance. Some databases might require periodic analysis and changing of their randomization characteristics in order to maintain best performance.

You can use the PSSR randomizer-performance reports to evaluate how well your randomizer is distributing your root segments. By running PSSR iteratively, you can judge potential DBD changes without having to create a new version of the database. After using the PSSR to determine better values for the DBD randomization parameters, you can reorganize your database using those parameters.

Example of steps for tuning analysis

This example uses DBD overrides (on the PSSR control statement in the CTL data set) to model potential DBD changes. PSSR is run iteratively, changing the DBD overrides on each iteration until a satisfactory randomization is obtained. The following steps are required:
1. “Unload the database with IMS HD Reorganization Unload utility or with an equivalent program”

2. “Run PSSR several times, each time with different DBD overrides”

Once the new DBD parameters are selected, the database should be reorganized as described in “Example 1: Normal reorganization (Changing root addressable area)” on page 212.

By monitoring the performance of your randomizing routines using PSSR (using PSSR, the HD Tuning Aid utility of IMS High Performance Pointer Checker, or both), you can detect conditions that might affect database performance. Proper analysis and tuning can then keep your databases in the best condition.

**Unload the database with IMS HD Reorganization Unload utility or with an equivalent program**

Unload the database with IMS HD Reorganization Unload utility or an equivalent program, such as IMS High Performance Unload for z/OS unload utilities.

In this scenario, the database is assumed to be unloaded as in “Example 1: Normal reorganization (Changing root addressable area)” on page 212 (see Figure 51 on page 212 for the JCL). It is a standard database unload execution.

**Run PSSR several times, each time with different DBD overrides**

This step prints the randomization reports. It consists of five executions of PSSR (see Figure 57 on page 221). The only difference between the five cases is the PSSR control statement.

1. This case uses the DBD without overriding any parameters. It represents the database condition that this analysis is trying to improve. All PSSR reports (including the potentially large Record Sizes report) are printed.

2. Here the number of blocks in the root addressable area is changed to 100 (column 20), and the number of RAPs in one block is changed to 4 (column 29). The leading zeros for these two control statement fields are required. The dash (-) in column 62 prevents the printing of both Record Sizes reports. These were printed in case 1 and there is no need to reprint them. (Modifying DBD parameters cannot change the size of a database record.)

3. The randomizing module is changed to DFSHDC30 (column 11), the number of blocks in the root addressable area is changed to 100 (column 20), and the number of RAPs in one block is changed to 4 (column 29). The Record Sizes reports are not printed (column 62).

4. The randomizing module is changed to DFSHDC20 (column 11), the number of blocks in the root addressable area is changed to 200 (column 20), and the number of RAPs in one block is changed to 1 (column 29). The Record Sizes reports are not printed (column 62).

5. The randomizing module is changed to DFSHDC10 (column 11), the number of blocks in the root addressable area is changed to 400 (column 20), and the number of RAPs in one block is changed to 3 (column 29). The Record Sizes reports are not printed (column 62).
/*---------------------------------------------------------------
//PHYBLK1 EXEC FABSPSSR
//STEPHD.CTL DD *
ORDHDM
/ *
//STEPHD.SYSIN DD *
SORT FIELDS=(5,20,BI,A),FILSZ=E9400000
MODS E15=(FABSRT15,50000,IMS2,N),E35=(FABSRT35,50000,IMS2,N)
END
/ *
/*---------------------------------------------------------------
//PHYBLK2 EXEC FABSPSSR
//STEPHD.CTL DD *
ORDHDM 0000100 004 - *
/ *
//STEPHD.SYSIN DD *
SORT FIELDS=(5,20,BI,A),FILSZ=E9400000
MODS E15=(FABSRT15,50000,IMS2,N),E35=(FABSRT35,50000,IMS2,N)
END
/ *
/*---------------------------------------------------------------
//PHYBLK3 EXEC FABSPSSR
//STEPHD.CTL DD *
ORDHDM DFSHDC30 0000100 004 - *
/ *
//STEPHD.SYSIN DD *
SORT FIELDS=(5,20,BI,A),FILSZ=E9400000
MODS E15=(FABSRT15,50000,IMS2,N),E35=(FABSRT35,50000,IMS2,N)
END
/ *
/*---------------------------------------------------------------
//PHYBLK4 EXEC FABSPSSR
//STEPHD.CTL DD *
ORDHDM DFSHDC20 0000200 001 - *
/ *
//STEPHD.SYSIN DD *
SORT FIELDS=(5,20,BI,A),FILSZ=E9400000
MODS E15=(FABSRT15,50000,IMS2,N),E35=(FABSRT35,50000,IMS2,N)
END
/ *
/*---------------------------------------------------------------
//PHYBLK5 EXEC FABSPSSR
//STEPHD.CTL DD *
ORDHDM DFSHDC10 0000400 003 - *
/ *
//STEPHD.SYSIN DD *
SORT FIELDS=(5,20,BI,A),FILSZ=E9400000
MODS E15=(FABSRT15,50000,IMS2,N),E35=(FABSRT35,50000,IMS2,N)
END
/ *

Figure 57. Example 3: DFSORT (PSSR) JCL
Example 4: Print reports about record sizes

PSSR produces two reports that describe the sizes of your database records.

These reports are:
- Record Sizes report
- Record Size Distribution report

To learn the size of every record in your database, print the Record Sizes report by coding ALL in column 65 as input for your control statement of the CTL data set (see “FABSPSSR CTL data set” on page 178 or “FABSPSSI CTL data set” on page 183).

Note: The Record Sizes report contains approximately 500 - 600 record sizes on each page. A large database could result in a very large printed report. For example, if your database has one million root segments, this report will be nearly 2000 pages long. Therefore, be sure that you really need to see this report (if your database is large) before coding ALL on your control statement.

One use for this report would be to locate the longest records. You might want to find and analyze those records to determine why they are so long. You can use PSSR and the IMS High Performance Pointer Checker for z/OS HD Pointer Checker utility and HD Tuning Aid utility and the IMS DBT Version 2 HD Tuning Aid utility to locate long records as follows:

1. Run PSSR to get the Record Sizes report. Suppose this report shows that the N-th database record is very long.
2. Run IMS High Performance Pointer Checker for z/OS HD Pointer Checker and HD Tuning Aid utilities to get the Actual Root per Block report. You can estimate which block contains the N-th root segment from this report.
3. Run IMS High Performance Pointer Checker for z/OS HD Pointer Checker utility to print a few database blocks surrounding your estimate of the block that contains the N-th root segment.
4. Extract the root segment keys from these blocks. Then, read each of those database records under DL/I to locate the longest records.

The Record Size Distribution report is usually more useful. It contains much of the same information as the Record Sizes report. However, the information in the Record Size Distribution report is in a compact, usable format. This report should help you select an appropriate byte limit and block size (or control interval size). It gives the average database record length in addition to a histogram of the distribution of all database record lengths. Byte limit and block size should usually be larger than the average database record length. This report can help you estimate how much data will go into overflow during a database reorganization (using particular values for byte limit and block size).

Example of steps for printing reports

In this example, PSSR is used to obtain information about database record sizes. No database reorganization is needed, so PSSR can be run without actually sorting the unloaded database data set. Eliminating the sorting reduces the amount of computer resources (both input/output time and processor time) needed to obtain the reports. The following steps are required:

1. “Unload the database with IMS HD Reorganization Unload utility or with an equivalent program” on page 223
2. “Run PSSR without sorting the unloaded database data set”

Unload the database with IMS HD Reorganization Unload utility or with an equivalent program

Unload the database with IMS HD Reorganization Unload utility or an equivalent program, such as IMS High Performance Unload for z/OS unload utilities.

In this scenario, the database is assumed to be unloaded as in “Example 1: Normal reorganization (Changing root addressable area)” on page 212 (see Figure 51 on page 212 for the JCL). It is a standard database unload execution.

Run PSSR without sorting the unloaded database data set

This step prints the Record Sizes reports. In the following example, the PSSR control statement is in the standard format used for printing both Record Sizes reports without sorting:

- The 0 in column 62 causes the actual sorting to be suppressed.
- Placing the word ALL in column 65 causes both the Record Sizes report and the Record Size Distribution report to be printed. If column 65 is left blank, only the Record Size Distribution report would be printed.

```
//DFSORT EXEC FABSPSSR
// DBDLIB='HPS.TEST.DBDLIB'
//STPHD.CTL DD *
ORDHDM 0 ALL */
//STPHD.SYSIN DD *
SORT     FIELDS=(5,20,BI,A),FILSZ=E9400000
        MODS E15=(FABSRT15,50000,IMS2,N),E35=(FABSRT35,50000,IMS2,N)
        END
/*
```

Figure 58. Example 4: DFSORT (PSSR) JCL
Example 5: HALDB processing

This example illustrates how to use PSSR to sort an entire database data set that is unloaded by the IMS HD Reorganization Unload utility. This method enables you to reorganize a HALDB.

In this example, it is assumed that there is no logical relationship in the database. The typical tasks are as follows:
1. Create an image copy of the old HALDB.
2. Unload the HALDB with the IMS HD Reorganization Unload utility or an equivalent program, such as IMS High Performance Unload for z/OS FABHURG1 (with *HD format) or FABHFSU (with UL format).
3. Create a new DBD. For HALDB, change the partition information using a partition definition utility if necessary.
4. Sort the unloaded database data set with PSSR, using the FABSPSSI procedure.
5. Delete the cluster for the old HALDB, and define a new one.
6. Reload the database with the IMS HD Reorganization Reload utility or an equivalent program, such as IMS HP Load.
7. Create an image copy of the new HALDB.
8. Run IMS High Performance Pointer Checker for z/OS HD Pointer Checker utility, using the image copy as input.

The PSSR JCL can be used to sort the unloaded data set into the SORTOUT data set:

```
//PSSRSORT EXEC FABSPSSI,
// IMSHDUN='UNLOAD.DATASET'
//STEPHD.SORTOUT DD DSN=SORTOUT.DATASET,DISP=NEW,
//   SPACE=(CYL,(10,1)),
//   UNIT=SYSDA,VOL=SER=000000,
//   DCB=UNLOAD.DATASET
//STEPHD.CTL DD *
PHDB01
/*
```
Part 5. Bitmap Resetter

The Bitmap Resetter utility adjusts bitmaps to accommodate denser packing of HD database blocks.

Topics:
- Chapter 15, “Introduction to the Bitmap Resetter utility,” on page 227
- Chapter 16, “Resetting bitmaps with the Bitmap Resetter utility,” on page 229
Chapter 15. Introduction to the Bitmap Resetter utility

The Bitmap Resetter utility can be used to adjust the bitmap of an HDAM, HIDAM, PHDAM, or PHIDAM database to accommodate denser packing of the database blocks.

A bit in the bitmap block indicates whether a block has free space, and the order of the bits in the bitmap block corresponds to the order of blocks.

Normally, when the bit is '1', it means that the corresponding block contains a contiguous free space large enough to put the largest segment in the data set group, and when the bit is '0', it means that it has no space for it.

The determination whether a block has free space is done as follows:

- Though there might be segments with various lengths, the largest segment in the data set group is used for determination.
- For a variable segment, the longest length is used for determination. Therefore, the actual segment length might be smaller than the maximum length defined in the DBD.
- For a compression segment, the original length (of uncompressed status) is used for determination.

Therefore, though the block might have enough free space for segments, if the bit is turned to '0' if it is determined that there is not enough space by its check, no segment is inserted into the block.

If you use the Bitmap Resetter utility, you can specify the threshold value for free space size whether the bit in bitmap block is set to '1' or '0'. That is, you can reset the bit so as to put the segment data into the contiguous free space, which is larger than the segment but smaller than the maximum segment size in the data set group. You can specify a separate value for up to 10 data set groups.

Important: Bitmap Resetter is a utility that changes the bitmap data. Therefore, it is recommended that you run image copy before you run Bitmap Resetter.
Chapter 16. Resetting bitmaps with the Bitmap Resetter utility

Use the following topics to learn how to run the Bitmap Resetter utility.

Topics:

- “Resetting bitmaps” on page 230
- “Bitmap Resetter JCL requirements” on page 231
- “Bitmap Resetter control statements” on page 234
- “Bitmap Resetter reports” on page 236
Resetting bitmaps

To run the Bitmap Resetter utility to reset bitmaps, you must code the EXEC statement and DD statements, and optionally control statements to control the behavior of the utility.

Procedure

1. Optional: Take an image copy of the database.
   The Bitmap Resetter utility changes the bitmap data. Therefore, it is recommended that you run image copy before you run the utility.
2. Prepare JCL for the Bitmap Resetter utility, and code the EXEC statement and DD statements to meet the JCL requirements of the Bitmap Resetter utility.
   For more information about the EXEC statement and DD statements, see "Bitmap Resetter JCL requirements" on page 231.
3. Optional: Code the control statements in the HPSIN data set.
   For more information about the control statements, see "Bitmap Resetter control statements" on page 234.
4. Submit the job.
5. Check the output reports and messages, and ensure that the job ran successfully.
   For more information about the reports, see "Bitmap Resetter reports" on page 236.

Example

The following figure shows a JCL example to run the Bitmap Resetter utility.

```plaintext
//BMPRST EXEC PGM=HPSBRSET,
//  PARM='TSTHDL1,DBRC=N'
//STEPLIB DD DISP=SHR,DSN=HPLD.SHPSLMD0
//  DD DISP=SHR,DSN=IMS.SDSFRESL
//IMS DD DISP=SHR,DSN=DBDLIB
//HPSOUT DD SYSOUT**
//HPSOUT2 DD SYSOUT**
//HPSIN DD *
BMSETDSG=(1000,3000,6000)
/*
```

Figure 59. Example JCL to use the BitmapResetter utility
**Bitmap Resetter JCL requirements**

To run a Bitmap Resetter job, code the EXEC statement and DD statements that satisfy the Bitmap Resetter JCL requirements.

The keywords for the JCL stream that are needed to run the Bitmap Resetter utility is shown in the following table.

*Table 27. DD statements and record format for Bitmap Resetter*

<table>
<thead>
<tr>
<th>DDNAME</th>
<th>Use</th>
<th>Format</th>
<th>Need</th>
</tr>
</thead>
<tbody>
<tr>
<td>STEPLIB</td>
<td>Input</td>
<td></td>
<td>Required See Note 1</td>
</tr>
<tr>
<td>IMS</td>
<td>Input</td>
<td></td>
<td>Required</td>
</tr>
<tr>
<td>HPSIN</td>
<td>Input</td>
<td>LRECL=80</td>
<td>Required</td>
</tr>
<tr>
<td>RECONx</td>
<td>Input</td>
<td></td>
<td>Optional for Non-HALDB</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Required for HALDB</td>
</tr>
<tr>
<td>database</td>
<td>Input/Output</td>
<td>Same as that used in the IMS control region startup JCL stream</td>
<td>Optional See Note 3</td>
</tr>
<tr>
<td>IMSDALIB</td>
<td>Input</td>
<td></td>
<td>Optional</td>
</tr>
<tr>
<td>HPSOUT</td>
<td>Output</td>
<td>LRECL=133</td>
<td>Required</td>
</tr>
<tr>
<td>HPSOUT2</td>
<td>Output</td>
<td>LRECL=133</td>
<td>Required</td>
</tr>
<tr>
<td>SYSUDUMP</td>
<td>Output</td>
<td>LRECL=133</td>
<td>Optional</td>
</tr>
</tbody>
</table>

**Notes:**
1. The load module library of STEPLIB must be authorized through Authorized Program Facility (APF).
2. If you use dynamic allocation, do not use the DD statements for these data sets.

**EXEC**  Must be in the following form:

```
//name EXEC PGM=HPSBRSET,PARM='dbname, DBRC=Y'
```

*dbname*  The name of the DBD.

*DBRC*=

Y  Requires DBD authorization. Specify DBRC=Y when you process HALDBs.

N  If DBRC=FORCE is set in SCD, returns HPSB0601E error. To use for a HALDB, you must set DBRC=Y.

**STEPLIB DD**

```
//STEPLIB DD DISP=SHR,DSNAME=pgmlib
// DD DISP=SHR,DSNAME=IMSESA.RESLIB
```

*pgmlib*  The name of the library that contains the load module of Bitmap Resetter.

**IMSESA.RESLIB**  The name of the library that contains the IMS load modules.

You must specify these two libraries. The load module libraries that are specified on the STEPLIB DD must be APF-authorized.
**IMSDLIB DD**

Specifies a partitioned data set or data sets that contain the dynamic allocation members for the database and the RECON data sets.

Allocation of the database and the RECON data sets is attempted in the following order:

1. The DD statements that are coded in the JCL stream
2. Dynamic allocation members in the IMSDALIB concatenation
3. Dynamic allocation members in JOBLIB or STEPLIB concatenation

For dynamic allocation of HALDB databases, the RECON data sets are used instead of the IMSDALIB DD statement.

**IMS DD**

Describes the library that contains the DBD referred to in the PARM field of the EXEC statement. This data set must reside on a direct-access device.

**HPSIN DD**

 Defines a data set that contains control statements that specify the functions of Bitmap Resetter. This DD statement is required. If the DD is omitted, the run terminates. For details about control statements, see “Bitmap Resetter control statements” on page 234.

**RECON1 DD**

Defines the first Database Recovery Control (DBRC) RECON data set.

**RECON2 DD**

Defines the second DBRC RECON data set.

**RECON3 DD**

Defines the optional DBRC RECON data set that is used when an error is encountered in RECON1 or RECON2. This RECON data set must be the same RECON data set as the one the control region is using.

**Note:** If you use dynamic allocation, these RECON data set ddnames must not be used.

**database DD**

Defines the database data set to be reset. You need one statement for each data set that appears in the DBD that describes this database. The ddname must match the ddname in the DBD. No DD statements are required for whatever primary index and secondary indexes that might be associated with this database. This data set must reside on a direct-access device.

**Note:** If you use dynamic allocation, do not use the DD statement for the database data sets.

For HALDBs, no DD statements are necessary for any database data sets because the data sets are always dynamically allocated.

**HPSOUT DD**

Reports, such as the HPSIN control statements report, Parameters and DBD/Data Sets report, error and warning messages, and Bitmap Reset Process Summary report, are printed. If the DD is omitted, the run terminates.

**HPSOUT2 DD**

Old and new contents of bitmap is printed. If the DD is omitted, no report is printed.
SYSUDUMP DD
Defines a dump data set. Most errors encountered end the Bitmap Resetter utility job by issuing an ABEND macro with the DUMP operand. Only a SYSUDUMP DD is needed. The SYSABEND DD statement is not required.
Bitmap Resetter control statements

Bitmap Resetter control statements specify the functions of Bitmap Resetter.

The following table lists the parameters that you can specify for the process control of the HPSIN data set.

Table 28. Bitmap Resetter control statements

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted values</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMSETDSG=(n1,n2,n3)</td>
<td>nnnnn or * or NO</td>
<td>NO</td>
<td>Specifies the threshold value for determining whether a block has space for DSG.</td>
</tr>
<tr>
<td>BMSETPART=(part_a,part_b,part_c)</td>
<td>nnnnnnnnn (7 digit)</td>
<td>N/A</td>
<td>When you select the partitions to be reset in the bitmap, use this optional control statement.</td>
</tr>
</tbody>
</table>

**BMSETDSG control statement**

Specifies the threshold value for determining whether a block has space for data set group (DSG).

```
BMSETDSG=(n1,n2,n3)
```

**NO**  Do not reset bitmap for this data set group.

*  Use the largest segment value for the data set group in the same manner as the IMS.

```
nnn
```

Use this value to specify the size to reset the bitmap block for the data set group. You can specify a value in the range of 6 - 32,000.

You can specify one of these parameters up to 10 times. When the number of specified parameters is less than the actual number of the data set groups, the bitmap resetter processes from the first to the specified number of the data set group.

When the number of specified parameters is more than number of the data set groups, the bitmap resetter processes from the first to the actual number of the data set groups.
**BMSETPART control statement**

When you select the partitions to be reset the bitmap, use this optional control statement. If you do not specify this control statement, and if the database is a HALDB, all partitions are processed.

```
BMSETPART=(partition)
```

*partition*

Specifies the partition name to be reset.

You can specify the value for the partition name in multiple BMSETPART control statements. When you specify multiple BMSETPART control statements, the statements do not need to be consecutive.

Example of a normal case:

```
BMSETPART=(PartA,PartB)
BMSETPART=(PartC,PartD)
```

You can specify up to 20 partitions at once.
Bitmap Resetter reports

The Bitmap Resetter utility generates several reports that contain information about the job and the bitmap blocks.

HPSIN Control Statements report

This report contains an echo back of the control statements that you specified in the HPSIN data set.

The following figure shows an example of the HPSIN Control Statements report.

Parameter and DBD/Data Sets report

This report provides information about the DBD definition and the parameter values that you specified.

The following figure shows an example of the Parameters and DBD/Data Sets report.
PARAMETER STRING
The parameters of the Bitmap Resetter program.

ANCHOR POINTS
The number of RAPs.

DATA SET GROUPS
The number of data set groups.

LRECL/CII
The size of block/CI. For a HALDB database, (PART) is displayed. To know the size of block/CI, see "Bitmap Reset Process Summary report."

MAXSEGML
The length of the maximum segment.

MINSEGML
The length of the minimum segment.

PARM VALUE
The parameter value of BMSETDSG that you specified.

**Bitmap Reset Process Summary report**

This report contains a summary of the bitmap reset process. For each data set, statistics about bitmap blocks are provided.

The following figure shows an example of the Bitmap Reset Process Summary report.

---

**Figure 62. Bitmap Reset Process Summary report**

**BITMAP BLK#**
The number of bitmap blocks.

**NUMBER OF BLOCKS WITH FREE SPACE**
The number of blocks/CIs which bitmap status is updated.

**ADDED**
The number of bitmap data that are turned on.

**REMOVED**
The number of bitmap data that are turned off.

**UNCHANGED**
The number of bitmap data unchanged.
If 'NO' is specified to BMSETDSG, the data of that data set group is dropped from this report. If NO is specified for BMSETDSG, other data set groups are not shown.

Error and Warning Messages report

This report contains error messages and warning messages that are issued during the job (if any).

The following figure shows an example of the Error and Warning Messages report.

---

**Old and New Contents of Bitmap report**

This report shows the beginning of the data set. You can use this report to compare the old and new bitmap information.

The following figure shows an example of the Old and New Contents of Bitmap report.

---

**Figure 63. Error and Warning Messages report**

**Figure 64. Old and New Content of Bitmap report**
Part 6. References

The following reference information helps you set site default values for IMS HP Load jobs and learn how to read syntax diagrams used in IMS HP Load information.

Topics:

• Chapter 17, “Setting installation default options,” on page 241
• Chapter 18, “How to read syntax diagrams,” on page 247
Chapter 17. Setting installation default options

You can change the default values for some options by creating an installation default option table (HPSROPTI) for your site.

Installation default option table is effective only for IMS HP Load. You cannot set installation default option values for the following utilities:

- Physical Sequence Sort for Reload
- Load API
- Bitmap Resetter

The following topics contain product-sensitive programming interface information. See the Programming interface information section at the end of this publication to understand the restrictions that are associated with this type of material.

Topics:
- “How the runtime options are determined” on page 242
- “Creating an installation default option table (HPSROPTI)” on page 243
- “HPSROPT macro statements” on page 244
How the runtime options are determined

Runtime options are determined by the parameter values that are set in the FRRIN data set, the installation default option table (HPSROPTI), and the system default option table (HPSROPTS).

The following figure illustrates how the values of the runtime options are determined. The sources that determine those values are placed in order of priority; that is, an option in a higher position in the figure overrides one in a lower position.

![Diagram illustrating how runtime options are determined]

Figure 65. How the runtime options are determined
Creating an installation default option table (HPSROPTI)

You can create an installation default option table (HPSROPTI) to change the default values of some options. If you link-edit HPSROPTI to your load module library of IMS HP Load, the default values specified in HPSROPTI apply to all IMS HP Load jobs that use the library.

Procedure

1. Copy the sample JCL stream HPSROPTG from the sample library.

   HPSROPTG is a sample JCL for use in creating an installation default option table (HPSROPTI). HPSROPTG is provided as a member of the HPS.SHPSSAMP library. HPSROPTG assembles the user-specified HPSROPT macro statements and link-edits the output of the assembly into the HPS.SHPSLMD0 load module library to build the module HPSROPTI.

2. Modify the HPSROPT macro statements in the copied JCL stream.

   The options that can be changed by creating an HPSROPTI are described in “HPSROPT macro statements” on page 244.

   Example:

   Use the following HPSROPT macro statements to change the default values as follows:

   • Change the default value of the CHKTRAIL option to (YES,ERROR).
   • Change the default value of the OADSPR option to YES.

   HPSROPT START
   HPSROPT RELOAD,CHKTRAIL=(YES,ERROR),OADSPR=YES
   HPSROPT END

3. Submit the JCL stream to assemble the HPSROPT macro statements and link edit the module HPSROPTI into the IMS HP Load load module library or a user library.
HPSROPT macro statements

The installation default options are specified by coding a set of HPSROPT macro statements.

These statements are assembler language macro instructions and therefore are subject to the rules contained in the High Level Assembler Programmer’s Guide.

The general syntax of the HPSROPT macro statement is as follows:
You must specify a set of HPSROPT macro statements in the following sequence:

1. An HPSROPT macro statement that specifies START as the first positional parameter.

2. An HPSROPT macro statement that specifies RELOAD as the first positional parameter, plus keywords and their parameters.
3. An HPSROPT macro statement that specifies END as the first positional parameter.

The option keywords that are supported for the RELOAD parameter are as follows:

- CHKEMPTYUL, CHKTRAIL, CHKUPDT, COMPAUTH, GDGBASE, GDGLIMIT, HPIO, ILDSBLD, LEDATA, ILESTORC, ITKBLOAD, ITKBSRVR, OADSPR, PAD, PARTINIT, REPAIRILK, SORT, TWINSTAT, and WF1REC.
Chapter 18. How to read syntax diagrams

The following rules apply to the syntax diagrams that are used in this information:

• Read the syntax diagrams from left to right, from top to bottom, following the path of the line. The following conventions are used:
  – The >>--- symbol indicates the beginning of a syntax diagram.
  – The ---> symbol indicates that the syntax diagram is continued on the next line.
  – The >--- symbol indicates that a syntax diagram is continued from the previous line.
  – The --->< symbol indicates the end of a syntax diagram.
• Required items appear on the horizontal line (the main path).
  ![Required Item Diagram]
• Optional items appear below the main path.
  ![Optional Item Diagram]
If an optional item appears above the main path, that item has no effect on the execution of the syntax element and is used only for readability.
  ![Optional Item Above Diagram]
• 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 main path.
  ![Required Choice Diagram]
If choosing one of the items is optional, the entire stack appears below the main path.
  ![Optional Choice Diagram]
If one of the items is the default, it appears above the main path, and the remaining choices are shown below.
  ![Default Choice Diagram]
• An arrow returning to the left, above the main line, indicates an item that can be repeated.
If the repeat arrow contains a comma, you must separate repeated items with a comma.

A repeat arrow above a stack indicates that you can repeat the items in the stack.

- Keywords, and their minimum abbreviations if applicable, appear in uppercase. They must be spelled exactly as shown. Variables appear in all lowercase italic letters (for example, column-name). They represent user-supplied names or values.
- Separate keywords and parameters by at least one space if no intervening punctuation is shown in the diagram.
- Enter punctuation marks, parentheses, arithmetic operators, and other symbols exactly as shown in the diagram.
- Footnotes are shown by a number in parentheses; for example, (1).
Part 7. Troubleshooting

Use the following topics to diagnose and correct problems that you experience with IMS High Performance Load.

Topics:
- Chapter 19, “Messages and codes,” on page 251
- Chapter 20, “Gathering diagnostic information,” on page 319
- Chapter 21, “Diagnostics Aid,” on page 321
Chapter 19. Messages and codes

This reference section provides detailed information about the abend codes, return codes, and messages that might be issued during the execution of IMS High Performance Load, Physical Sequence Sort for Reload, and Bitmap Resetter.

Topics:

- “How to look up message explanations” on page 252
- “Abend codes” on page 253
- “Return codes” on page 254
- “Messages” on page 258
How to look up message explanations

You can use several methods to search for messages and codes.

**Searching for messages on the web**

You can use any of the popular search engines that are available on the web to search for message explanations. When you type the specific message number or code into the search engine, you will be presented with links to the message information in IBM Knowledge Center.
Abend codes

IMS HP Load issues an abend code when a module of IMS HP Load terminates abnormally.

Every \textit{nnnn} abend code is accompanied by an HPSR\textit{nnnn}E, FABS\textit{nnnn}E, or HPSB\textit{nnnn}E message. (\textit{nnnn} is a four-digit number that identifies the abend code and message.) See the message description for the 3\textit{nnn} and 4000 abend code.

For the Load API, the abend code is U4000. For the problem analysis of the Load API, see the HPSR\textit{nnnn}E messages that are issued before the HPSR4000E message. For other messages of IMS HP Load, see “Messages” on page 258.
Return codes

The following topics explain the return codes that are issued by IMS High Performance Load, Physical Sequence Sort for Reload, and Bitmap Resetter.

IMS HP Load return codes

When IMS HP Load is used in the IMS Database Reorganization Expert environment, the return codes in the following table are returned.

Table 29. Return codes of IMS HP Load

<table>
<thead>
<tr>
<th>Return code</th>
<th>Reason code</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>None.</td>
<td>Successful completion.</td>
</tr>
</tbody>
</table>

254 IMS High Performance Load User’s Guide
Table 29. Return codes of IMS HP Load (continued)

<table>
<thead>
<tr>
<th>Return code</th>
<th>Reason code</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>4 (see Note)</td>
<td>X'0001'</td>
<td>No header or trailer record. This code is returned if the input unloaded data set did not have an unload type header or trailer record. The database was successfully reloaded. IMS HP Load issues a message that explains which type of record was missing.</td>
</tr>
<tr>
<td></td>
<td>X'0002'</td>
<td>A return code 4 is received from the HDAM randomizer. The database is successfully reloaded except for the root segment and its dependent segments.</td>
</tr>
<tr>
<td></td>
<td>X'0004'</td>
<td>Some segments are saved in the SEQERROR data set by the SEQERROR=SAVE option. Those segments are available for review and reinsertion by a user-written program.</td>
</tr>
<tr>
<td></td>
<td>X'0008'</td>
<td>Although the database reloaded is involved in a logical relationship or has a secondary index, DFSURWF1 DD DUMMY is specified.</td>
</tr>
<tr>
<td></td>
<td>X'0010'</td>
<td>Although SEQERROR=SAVE is specified in the FRRIN DD statement, SEQERROR DD DUMMY is specified.</td>
</tr>
<tr>
<td></td>
<td>X'0020'</td>
<td>An inconsistent return code is received from the exit routine that is specified on the USERHDRX control statement.</td>
</tr>
<tr>
<td></td>
<td>X'0040'</td>
<td>No segment is read from the non-dummy input (DFSUINPT). The database is empty. Tip: The return code can be changed by specifying the CHKEMPTYUL statement. For more information, see “CHKEMPTYUL control statement” on page 50.</td>
</tr>
<tr>
<td></td>
<td>X'0200'</td>
<td>Although the reloaded database is involved in a secondary index, HPSRSIDXDD DUMMY is specified.</td>
</tr>
<tr>
<td></td>
<td>X'0400'</td>
<td>An update between unload and reload processes is found, but processing continued due to the specification of CHKUPDT=YES,WARN.</td>
</tr>
<tr>
<td></td>
<td>X'0800'</td>
<td>One or more errors occurred in the IMS Tools Knowledge Base Output repository processing.</td>
</tr>
<tr>
<td></td>
<td>X'1000'</td>
<td>API function error in IMS Library Integrity Utilities (LIU), Integrity Checker.</td>
</tr>
</tbody>
</table>
|             | X'2000'     | Although partition initialization function is used, some partitions were not initialized by one of the following reasons:  
  • The flag in the RECON shows that the partition does not need to be initialized (NOPINIT).  
  • The partition information is not found in the unloaded data set.  
  • Although PARTINIT=YES,LPART is specified, one or more partitions are not specified on the LPART statement.) |
|             | X'4000'     | The DB Sensor component returned an error code. |
|             | X'8000'     | Although REPAIRILK=YES is specified, some partitions are not reloaded. When you specify REPAIRILK=YES, ensure that the entire HALDB (that is, all the partitions of the HALDB) is reorganized with the REPAIRILK=YES option to avoid problems that might be caused by duplicate segment ILKs. |
Table 29. Return codes of IMS HP Load (continued)

<table>
<thead>
<tr>
<th>Return code</th>
<th>Reason code</th>
<th>Meaning</th>
</tr>
</thead>
</table>
| 8           | None.       | This return code is returned in one of the following cases:  
  • An incorrect control statement was encountered. IMS HP Load issues a message that the user-supplied control statement contained an error.  
  • A library of STEPLIB is not APF-authorized. The load module library of IMS HP Load must be APF-authorized.  
  The database was not reloaded. |
| 16          | None.       | The database was not reloaded successfully. The reload segment count was not equal to the unload count. Or the reload segment count cannot be verified because a trailer record is not found at the end of the unload data set when the unload data set should have a trailer record. |
| 100–999     | None.       | A user exit routine requested to stop the processing during the reloading process and to return the return code. |

Note: For return code 4, the reason code is a logical sum of the reason codes that are listed in this table.

Physical Sequence Sort for Reload return codes

In the MVS batch environment, PSSR consists of DFSORT exit routines. The return code that is returned to the operating system comes from DFSORT (not from PSSR).

The DFSORT return codes for a PSSR run, and their meanings, are summarized in the following table.

Table 30. Return codes of Physical Sequence Sort for Reload (DFSORT)

<table>
<thead>
<tr>
<th>Return code</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>DFSORT is successfully executed and the sorting ends. This code is the normal return code for all PSSR runs in which column 62 of the PSSR control statement does not contain 0.</td>
</tr>
<tr>
<td>16</td>
<td>If column 62 of the PSSR control statement contains 0, DFSORT ends. This is not an error. Otherwise, DFSORT encounters an error during execution and ends.</td>
</tr>
<tr>
<td>20</td>
<td>The SYSOUT DD statement is missing.</td>
</tr>
</tbody>
</table>

In the IMS environment, PSSR returns the reason codes summarized in the following table.

Table 31. Return codes of Physical Sequence Sort for Reload in IMS environment

<table>
<thead>
<tr>
<th>Return code</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>This code is the normal return code for all PSSR runs in which column 62 of the PSSR control statement does not contain 0.</td>
</tr>
</tbody>
</table>
| 4           | • Some segments are not stored in the output data set. The error reason is shown in message FABSnnnW.  
  • Some specifications to the PSSR CTL statement are invalid. PSSR ignores them or regards them as blanks.  
  • The specified RAA size is larger than or equal to 4 GB. |
Table 31. Return codes of Physical Sequence Sort for Reload in IMS environment (continued)

<table>
<thead>
<tr>
<th>Return code</th>
<th>Meaning</th>
</tr>
</thead>
</table>
| 16          | If column 62 of the PSSR control statement contains 0, DFSORT ends. *This is not an error.*  
|             | Otherwise, DFSORT encounters an error during execution and ends. |
| 20          | The SYSOUT DD statement is missing. |
| 100–999     | Indicates termination by a user exit. PSSR ends the process and passes this return code back as the job return code. The return code value can be in the range of 100 - 999. |

**Bitmap Resetter return codes**

Bitmap Resetter returns the return codes that are shown in the following table.

Table 32. Return codes of Bitmap Resetter

<table>
<thead>
<tr>
<th>Return code</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Successfully completed.</td>
</tr>
<tr>
<td>4</td>
<td>Warning. An error except for the following (RC=8, 12) cases has occurred.</td>
</tr>
</tbody>
</table>
| 8           | The case that there is no specification of database data set IMSDALIB or STEPLIB DD.  
|             | One or more errors occurred. The error messages are reported in the HPSIN Control Statements report or in the Error and Warning Messages report. |
| 12          | The following error occurred: HPSB0103E, HPSB0104E, HPSB0105E, HPSB0108E, HPSB0110E, HPSB0122E, and HPSB3unnE other than HPSB3999E. |
| 16          | One or more errors occurred, and the processing aborted. |
Messages

Use the information in these messages to help you diagnose and solve IMS HP Load problems.

Message format

All the messages that are issued by IMS HP Load have the following format:

HPSRnnnxx
FABSnnnxx
HPSBnnnxx

Where:

HPSR Indicates that the message is issued by IMS HP Load
FABS Indicates that the message is issued by Physical Sequence Sort for Reload
HPSB Indicates that the message is issued by Bitmap Resetter
nnn Indicates the four-digit message identification number
x Indicates the severity of the message:
  E Indicates that an error occurred, which might or might not require operator intervention.
  I Indicates that the message is informational only.
  W Indicates that the message is a warning to alert you to a possible error condition.

Each message also includes the following information:

Explanation:
The Explanation section explains what the message text means, why it occurred, and what its variables represent.

System action:
The System action section explains what the system will do in response to the event that triggered this message.

User response:
The User response section describes whether a response is necessary, what the appropriate response is, and how the response will affect the system or program.
**IMS HP Load messages**

These topics provide HPSR messages which are issued by IMS HP Load.

---

<table>
<thead>
<tr>
<th>HPSR0001I</th>
<th>RELOAD FUNCTION ENDED, HIGHEST RETURN CODE IS return_code (REASON CODE: reason_code)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation:</strong> IMS HP Load ended. The return code and the reason code show the result of the processing. For the description of each return code and reason code, see <a href="#">“IMS HP Load return codes” on page 254</a>.</td>
<td></td>
</tr>
<tr>
<td><strong>System action:</strong> IMS HP Load sets the indicated return code and continues processing the HD unloaded data set.</td>
<td></td>
</tr>
<tr>
<td><strong>User response:</strong> None. This message is informational.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>HPSR0002W</th>
<th>FIRST RECORD WAS NOT A HDR RECORD</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation:</strong> The first record in the HD unloaded data set (DFSUINPT DD) was not an HD Unload utility header record.</td>
<td></td>
</tr>
<tr>
<td><strong>System action:</strong> IMS HP Load sets a return code of 4 and continues processing the HD unloaded data set.</td>
<td></td>
</tr>
<tr>
<td><strong>User response:</strong> Ensure that the input HD unloaded data set is correct. If it is not, correct it, and resubmit the job.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>HPSR0003W</th>
<th>LAST RECORD WAS NOT A TRAILER RECORD</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation:</strong> An unloaded data set must have a trailer record as its last record, though the last record in the HD unloaded data set (DFSUINPT DD) was not an unload trailer record.</td>
<td></td>
</tr>
<tr>
<td><strong>System action:</strong> IMS HP Load sets a return code of 4 and completes the reloading of the database.</td>
<td></td>
</tr>
<tr>
<td><strong>User response:</strong> Ensure that the input HD unloaded data set is correct. If it is not, correct it, and resubmit the job.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>HPSR0004E</th>
<th>UNLOAD AND RELOAD SEGMENT COUNTS ARE NOT THE SAME</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation:</strong> The number of records in the HD unloaded data set is not equal to the number of records that are written into the reloaded database.</td>
<td></td>
</tr>
<tr>
<td><strong>System action:</strong> IMS HP Load sets a return code of 16.</td>
<td></td>
</tr>
<tr>
<td><strong>User response:</strong> Contact IBM Software Support.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>HPSR0005E</th>
<th>INVALID CONTROL STATEMENT WAS Encountered</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation:</strong> The user-supplied control statement is found to contain an error.</td>
<td></td>
</tr>
</tbody>
</table>

**System action:** IMS HP Load sets a return code of 8. **User response:** Correct the error and rerun the job.

<table>
<thead>
<tr>
<th>HPSR0006W</th>
<th>RETURN CODE 4 FROM HDAM RANDOMIZER, SEGMENTS DROPPED</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation:</strong> IMS HP Load received a return code of 4 from the HDAM randomizer. The root segment and its dependent segments were not reloaded. This message is followed by the data of the segments.</td>
<td></td>
</tr>
<tr>
<td><strong>System action:</strong> IMS HP Load continues processing and sets a return code of 4. In the case of the Load API processing, the return code is 0.</td>
<td></td>
</tr>
<tr>
<td><strong>User response:</strong> Check the reason of the return code 4 from the HDAM randomizer. Verify the logic of the HDAM randomizer. Correct it, and rerun the job if necessary.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>HPSR0007W</th>
<th>DUMMY SPECIFIED FOR DFSURWF1 DD</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation:</strong> The DFSURWF1 data set is used as input to resolve a logical or secondary index relationship. DFSURWF1 DD DUMMY is specified, although the database reloaded is involved in a logical relationship or with a secondary index.</td>
<td></td>
</tr>
<tr>
<td><strong>System action:</strong> IMS HP Load sets a return code of 4. In the case of the Load API processing, the return code is 0, and completes reloading the database.</td>
<td></td>
</tr>
<tr>
<td><strong>User response:</strong> Check the DBD and the PSB to see if logical or secondary index relationships must be resolved. If they must be resolved, specify the DFSURWF1 data set, and rerun the job.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>HPSR0009W</th>
<th>DUMMY SPECIFIED FOR SEQERROR DD</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation:</strong> Although SEQERROR=SAVE is specified in the FRRIN or HPSIN DD statement, SEQERROR DD DUMMY is specified. The SEQERROR data set is required for saving segments in key sequence errors.</td>
<td></td>
</tr>
<tr>
<td><strong>System action:</strong> IMS HP Load continues processing and sets a return code of 4.</td>
<td></td>
</tr>
<tr>
<td><strong>User response:</strong> Ensure that a DD statement is present for SEQERROR indicated, and that it properly identifies the correct data set. Specify the SEQERROR data set, and rerun the job if necessary.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>HPSR0010W</th>
<th>NO SEGMENT WAS READ FROM INPUT</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explanation:</strong> IMS HP Load attempted to reload a...</td>
<td></td>
</tr>
</tbody>
</table>
database from the non-dummy input (DFSUINPT), but no segment is read from the input.

**System action:** IMS HP Load sets a return code of 4 and ends. The reloaded database is empty.

**User response:** Ensure that the correct DDB and the unloaded data (DFSUINPT) are used for the processing. If not, correct the error and rerun the job.

**Problem determination:** Check the unloaded data format that is determined by IMS HP Load. The format is reported in the FRRIN or HPSIN Control Statements report. If the reported format is not the one you expected, the input data might be corrupted.

---

**HPSR0011W NO HEADER OR TRAILER RECORD EXISTED**

**Explanation:** Although the exit routine that is specified on the USERHDRX control statement has set the indicator, which means “there is a header or a trailer record”, it did not return a return code of 4.

**System action:** IMS HP Load sets a return code of 4 and ends.

**User response:** Check the routine and the input data set (DFSUINPT). If the input data set has a header or a trailer record, the routine should set the indicator and return a return code of 4. If it does not, it should not set the indicator nor return a return code of 4.

---

**HPSR0012W DBD dbname IS USING DATA SETS NOT REGISTERED WITH DBRC**

**Explanation:** DBRC is called to create a REORG record for the database, but one or more data set names are not registered with DBRC.

**System action:** IMS HP Load sets a return code of 4. In the case of the Load API processing, the return code is 0.

**User response:** Ensure that the correct RECON data sets are used. Correct the error, and rerun the job.

---

**HPSR0013W DUMMY SPECIFIED FOR HPSRSIDX DD**

**Explanation:** The HPSRSIDX data set is used as input to resolve the secondary index relationship. Although the database that was reloaded is involved in a secondary index, HPSRSIDX DD DUMMY is specified.

**System action:** IMS HP Load sets a return code of 4. In the case of the Load API processing, the return code is 0. And completes reloading the database.

**User response:** Check the DBD and the PSB and identify if they are required to resolve a secondary index relationship. If it is necessary to resolve a secondary index relationship, specify one of the following data sets and rerun the job:

- HPSRSIDX data set
- DFSURWF1 data set, but do not specify the HPSRSIDX data set

---

**HPSR0015I RELOAD TASK FORCED TO END DUE TO ERROR IN ANOTHER TASK**

**Explanation:** The reload task of the IPR Driver was forced to end because of an error in another task.

**System action:** The reload task of the IPR Driver is forced to end.

**User response:** Identify the task that originated the error. Correct the error and rerun the job.

---

**HPSR0016I COMPRESSED SEGMENTS WERE NOT DECOMPRESSED AT UNLOAD. COMPRESS=N IS FORCED**

**Explanation:** COMPRESS=Y is specified in the FRRIN or HPSIN DD. The IMS HP Load utility must be run with COMPRESS=N.

**System action:** COMPRESS=N is forced, and processing continues.

**User response:** None. This message is informational.

---

**HPSR0020I DBD "dbname" IS PARTITIONED DB**

**Explanation:** The database to be reloaded is a partitioned database.

**System action:** Processing continues.

**User response:** None. This message is informational.

---

**HPSR0023I PARTITION LOAD FOR "partname" STARTED**

**Explanation:** Loading of the partition partname has just started.

**System action:** Processing continues.

**User response:** None. This message is informational.

---

**HPSR0024I PARTITION LOAD FOR "partition" COMPLETED**

**Explanation:** Loading of the indicated partition is complete.

**System action:** Processing continues. If no other partition is left, processing ends.

**User response:** None. This message is informational.

---

**HPSR0025I DBDSNAP=Y IS IGNORED**

**Explanation:** DBDSNAP=Y is specified in the FRRIN or HPSIN DD, but the SNAP DD is not specified or is DUMMY.

**System action:** Processing continues. IMS HP Load
utility does not print the dump of DBD for user reference.

User response: Specify SNAP DD, and rerun the job if necessary.

HPSR0026E BUFFERNO= AND DBRLBUF= KEYWORDS ARE MUTUALLY EXCLUSIVE

Explanation: Both BUFFERNO and DBRLBUF keywords are specified in the FRRIN or HPSIN data set. These keywords are mutually exclusive.

System action: Processing stops.

User response: Correct the error and rerun the job.

HPSR0027I nnn,nnn,nnn DB RECORDS PROCESSED
 - DBD=database PART=partition

Explanation: nnn,nnn,nnn database records are processed for the indicated database. PART=partition is displayed only for a partitioned database; database records are counted within each partition. This information message helps you to monitor the IMS HP Load job progress.

System action: IMS HP Load continues processing.

User response: None. This message is informational.

HPSR0028I nnn,nnn,nnn DB RECORDS PROCESSED
 - PART=pppppppp

Explanation: nnn,nnn,nnn database records are processed. PART=pppppppp is displayed only for a partitioned database. This information message helps you to monitor the progress of an IMS HP Load job, when some sequence error segments are reloaded. A preceding message or messages indicate the reason.

System action: IMS HP Load continues processing.

User response: None. This message is informational.

HPSR0029E LAPIPCBNO=, LAPIPCB=, AND LAPIPDBD= ARE MUTUALLY EXCLUSIVE

Explanation: Multiple control statements are specified among LAPIPCBNO=, LAPIPCB=, and LAPIPDBD= in the FRRIN data set. They are mutually exclusive.

System action: Processing stops.

User response: Correct the error and rerun the job.

HPSR0032W SOME SEGMENTS WERE SAVED IN SEQERROR DD

Explanation: There were some key sequence errors in the unloaded data set. IMS HP Load saved segment in error and their dependent segments in the SEQERROR data set, as specified by SEQERROR=SAVE.

System action: IMS HP Load continues processing and sets a return code of 4. The number of segments that are saved is reported in the Segment Count Summary report.

User response: None. You can use the SEQERROR data set to insert segments with your user-written DL/I program.

HPSR0033I A SEGMENT IS OUT OF SEQUENCE
 - SEGMENT="segment_name"
 - FIELD="field_name"

Explanation: IMS HP Load attempted to put a segment chain whose sequence field is less than the previous segment on a twin chain. IMS HP Load reloads a database record in descending order.

System action: IMS HP Load continues processing.

User response: None. This message is informational.

HPSR0034I A SEQUENCE FIELD IS NOT UNIQUE
 - SEGMENT="segment_name"
 - FIELD="field_name"

Explanation: IMS HP Load attempted to put a segment segment_name whose sequence field field_name was not unique on a twin chain. But the specification of the sequence field is unique.

System action: IMS HP Load continues processing.

User response: None. This message is informational.

HPSR0036E LAST RECORD WAS NOT A TRAILER RECORD

Explanation: An unloaded data set must have a trailer record as its last record, though the last record in the HD unloaded data set (DFSUINPT DD) was not an unload trailer record.

System action: IMS HP Load sets a return code of 16.

User response: Verify that the input HD unloaded data set is correct. If it is not, correct it, and resubmit the job.

HPSR0037I DFSURWF1 RECORD CREATION IS SUPPRESSED

Explanation: Though the database to be reloaded has a logical relationship or has a secondary index, creation of the DFSURWF1 records is suppressed because WF1REC=NO is specified.
System action: IMS HP Load continues processing without producing DFSURWF1 records.
User response: If you need the DFSURWF1 data set, remove WF1REC=NO and rerun the job.

**HPSR0038I** INSTALLATION DEFAULT OPTIONS TABLE HPSROPTI (LV,mn) IS FOUND

Explanation: The Installation Default Options Table (HPSROPTI) is found. HPSROPTI is generated with the HPSROPT macro of level mn. The default values in this table override the values in the System Default Option Table (HPSROPTS).

System action: IMS HP Load continues processing.
User response: None. This message is informational.

**HPSR0039I** GDG BASE ENTRY WAS CREATED

Explanation: Because GDGBASE=YES is specified, a GDG base entry was created by the IDCAMS command.

System action: IMS HP Load continues processing.
User response: None. This message is informational.

**HPSR0040I** THE AMOUNT OF DATABASE BUFFERS ADJUSTED

Explanation: IMS HP Load adjusted the amount of database buffers that are specified by the DBRLBUF statement or the BUFFERNO statement. The amount of database buffers adjusted can be calculated by using the following formula:

\[
\text{The amount of the access method buffers} = \text{specified number} + 10 \text{ tracks}
\]

Note: Specified by the DBIOBUF statement or JCL.

System action: The amount of database buffers is adjusted to the correct size, and IMS HP Load continues processing.
User response: None. This message is informational.

**HPSR0041I** GDG BASE ENTRY WAS NOT CREATED BECAUSE GDG BASE ENTRY ALREADY EXISTS

Explanation: GDGBASE=YES is specified. However, GDG base entry is not created because a GDG base entry with the name that is specified on the WF1DSN control statement exists.

System action: IMS HP Load continues processing.
User response: None. This message is informational.

**HPSR0042I** APF AUTHORIZATION CHECK FAILED (DD: ddname, REASON: reason_code)

Explanation: The APF-authorization check module (HPSRAPF0) returned an error for the ddname DD statement. The reason code reason_code is used internally by the module.

System action: Processing continues.
User response: This error is likely an internal system error. Contact IBM Software Support.

**HPSR0043I** WF1DSN IS IGNORED BECAUSE MULTIPLE DATABASES WERE PROCESSED

Explanation: The specification of the WF1DSN statement was ignored because multiple PCBs are specified in the PSB and a logical relationship or a secondary index is defined in those PCBs. In this case, Load API cannot allocate DFSURWF1 data sets dynamically.

System action: IMS HP Load continues processing.
User response: None. This message is informational.

**HPSR0044W** WF1DSN SPECIFICATION IS IGNORED BECAUSE DFSURWF1 DD IS SPECIFIED

Explanation: The specification of the WF1DSN statement was ignored because the DFSURWF1 DD statement is specified in the JCL stream.

System action: IMS HP Load continues processing. IMS HP Load uses the data set that is specified on the DFSURWF1 DD statement.
User response: If you want to use the data set that is specified on the WF1DSN statement, remove the DFSURWF1 DD statement and rerun the job.

**HPSR0045I** WFPHLQ IS IGNORED BECAUSE PSINDEX IS NOT DEFINED

Explanation: The WFPHLQ control statement is ignored because PSINDEX is not defined to the database that is to be reloaded.

System action: IMS High Performance Load ignores the parameter, and continues processing.
User response: None. This message is informational.

**HPSR0050E** UNABLE TO PROCESS THE SPECIFIED PCB (REASON: reason)

Explanation: The PCB that is specified with the control statements cannot be processed. The string reason shows the reason.
GSAMPCB  | The specified PCB is a GSAM PCB.
PROCOPT  | The PROCOPT parameter on the PCB statement specifies an incorrect code or combination of codes. The allowable codes are L and LS.
NOTFOUND | The specified PCB does not exist.

For the PCB name, see the HPSR4000E message that is issued together with this message.

System action: IMS HP Load issues a user 4000 abend.
User response: Correct the specification of the control statement and rerun the job.

HPSR0060E  | AN ERROR OCCURRED WHILE THE PIPES WERE PROCESSED
            | FUNC=NNNN
Explanation: An error occurred while the IPRPIPE service was processing. The following are the explanations of 'FUNC=NNNN':
INIT      | An error occurred when PIPE was allocated.
TERM      | An error occurred when PIPE was deallocated.
System action: IMS HP Load issues a user 4000 abend.
User response: See the user response sections of the messages that are issued together with this message.

HPSR0075E  | THE USER EXIT ROUTINE MUST BE AMODE=31.
Explanation: The user exit routine must be AMODE=31 for the Load API processing.
System action: IMS HP Load issues a user 4000 abend.
User response: Correct the error and rerun the job.

HPSR0100W  | "keyword" IS NOT APPLIED TO db_type DB.
            | SPECIFICATION IGNORED
            | RAAFORMAT=
            | SEARCH=
            | FRSPC=
            | RAPCHAIN=
            | DBRECDIST=
            | SORT=
            | SORTHIDAM=
Explanation: Keyword is not applied to the database to be reloaded because the database organization of the database is db_type. This parameter specification is ignored for this run.
System action: Processing continues; the parameter specification is ignored.
User response: Ensure that the control statement is correct and whether the correct DBD was used for this job. If necessary, correct the control statement in error and rerun the job.

HPSR0104E  | "parm_value" IS INVALID FOR "parm_name" PARAMETER
Explanation: An incorrect control statement is detected in the control statement syntax checking.
System action: Processing stops.
User response: Correct the error and rerun the job.

HPSR0105W  | HPIO= IGNORED BECAUSE OF THE DEFINITION FOR IPR DRIVER
Explanation: The HPIO= specification for the reload task was ignored because the definition for IPR Driver overrides the specification.
**HPSR0106E • HPSR0115E**

**HPSR0106E NUMBERS IN DBRECDIST STATEMENTS MUST BE SPECIFIED IN ASCENDING ORDER**

**Explanation:** The range values that are specified in the DBRECDIST control statements must not be duplicated and must be in ascending order.

**System action:** Processing stops.

**User response:** Correct the error and rerun the job.

**HPSR0107E MORE THAN 20 RANGE VALUES ARE SPECIFIED**

**Explanation:** More than 20 range values are specified in the DBRECDIST control statements.

**System action:** Processing stops.

**User response:** Reduce the number of values that are specified on the DBRECDIST control statement.

**HPSR0108E INVALID DBRECDIST STATEMENT IS SPECIFIED**

**Explanation:** IMS HP Load detected an incorrect DBRECDIST control statement.

**System action:** Processing stops.

**User response:** If you do not want to generate the Distribution of Database Record Sizes report, specify 0 on the DBRECDIST control statement. If you want to generate it, specify up to 20 values, each of which must be a decimal number in the range of 1 - 9999999, on the DBRECDIST control statement.

**HPSR0109E IPR DRIVER NOT IN REQUIRED LEVEL (level)**

**Explanation:** The release level or maintenance level of the IPR Driver is lower than the level that is required for IMS HP Load to work correctly.

**System action:** Reload task ends with a return code of 8.

**User response:** Apply the required maintenance of the IPR Driver.

**HPSR0110W keyword IS IGNORED UNDER IMS REGION CONTROLLER**

**Explanation:** The keyword cannot be used under IMS region controller (PGM=DFSRRC00). The keyword parameter is ignored.

**System action:** Processing continues; the keyword parameter is ignored.

**User response:** To use the keyword, specify PGM=HPSRC00 and APF-authorize all the libraries that are in the STEPLIB DD concatenation.

**HPSR0111E USERHDR=, USERHDRX=, AND SORT= KEYWORDS ARE MUTUALLY EXCLUSIVE**

**Explanation:** All USERHDR=, USERHDRX=, and SORT= keywords are specified in the FRRIN or HPSIN data set. These keywords are mutually exclusive when processing HALDBs.

**System action:** Processing stops.

**User response:** Correct the error and rerun the job.

**HPSR0112E DUPLICATE ELEMENT (element) SPECIFIED IN USERHDR CONTROL STATEMENT**

**Explanation:** The same element element is specified in the USERHDR control statement. You can specify element only once in the control statement.

**System action:** Processing stops.

**User response:** Correct the parameter error, and rerun the job.

**HPSR0113E element MUST PRECEDE ROOTKEY, SEGKEY, OR SEGCKEY**

**Explanation:** The element that you specified (HDRLEN, SEGNAME, or SEGCODE) must precede one of the ROOTKEY, SEGKEY, or SEGCKEY elements.

**System action:** Processing stops.

**User response:** Correct the parameter error, and rerun the job.

**HPSR0114E MORE THAN 20 ELEMENTS ARE SPECIFIED IN USERHDR**

**Explanation:** More than 20 elements are specified on the USERHDR control statements.

**System action:** Processing stops.

**User response:** Reduce the number of elements on the USERHDR control statements, and rerun the job.

**HPSR0115E HDRLEN MUST BE SPECIFIED WITH SEGKEY, SEGCKEY, OR ROOTKEY**

**Explanation:** If the SEGKEY, SEGCKEY, or ROOTKEY element is specified, the HDRLEN element must be specified.

**System action:** Processing stops.

**User response:** Correct the parameter error, and rerun the job.
HPSR0116E  SEGNAME OR SEGCODE IS REQUIRED FOR SPECIFICATION OF USERHDR

**Explanation:** The USERHDR statement requires either the SEGNAME or the SEGCODE keyword, but neither keyword was provided.

**System action:** Processing continues with a return code of 8.

**User response:** Provide the SEGNAME or the SEGCODE keyword and rerun the job.

---

HPSR01191  PHYSICAL SEQUENTIAL SORTING PROCESSED (reason)

**Explanation:** SORT=COND is specified and physical sequential sorting is processed by one of the following reasons:

<table>
<thead>
<tr>
<th>Reason</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHANGE DB ORGANIZATION</td>
<td>Database organization is changed from (P)HDAM to (P)HDAM.</td>
</tr>
<tr>
<td>CHANGE PARTITION</td>
<td>The number of partitions has been changed since the unload process.</td>
</tr>
<tr>
<td>CHANGE RANDOMIZER</td>
<td>The randomizing module name that is used for reload has been changed since the unload process.</td>
</tr>
<tr>
<td>CHANGE RAP#</td>
<td>RAP number that is used for reload has been changed since the unload process.</td>
</tr>
<tr>
<td>CHANGE RBN</td>
<td>The maximum BLK/CI number in RAA that is used for reload has been changed since the unload process.</td>
</tr>
<tr>
<td>CHANGE ROOT KEY LENGTH</td>
<td>The length of the root key that is used for reload has been changed since the unload process.</td>
</tr>
<tr>
<td>CHANGE ROOT KEY POSITION</td>
<td>The starting position of root key that is used for reload has been changed since the unload process.</td>
</tr>
<tr>
<td>CONCATENATED UNLOAD D/S</td>
<td>The input unloaded data sets are concatenated.</td>
</tr>
<tr>
<td>CS HEADER</td>
<td>The input unloaded data set was created in *CS format.</td>
</tr>
<tr>
<td>Fallback to non-HALDB</td>
<td>Database organization is changed from HALDB to non-HALDB.</td>
</tr>
<tr>
<td>LOAD API</td>
<td>The Load API function is used.</td>
</tr>
<tr>
<td>LOW LEVEL UNLOAD D/S</td>
<td>The input unloaded data set was not created by IMS HP Unload with the CHECKREC=YES option, or the release level or the maintenance level of IMS HP Unload is lower than the level that is required for IMS HP Load to work correctly.</td>
</tr>
<tr>
<td>MIGRATION TO HALDB</td>
<td>Database organization is changed from non-HALDB to HALDB.</td>
</tr>
<tr>
<td>UNLOADED VIA SECONDARY INDEX</td>
<td>The input unloaded data set was created by referring to the secondary index database.</td>
</tr>
<tr>
<td>USER HEADER</td>
<td>USERHDR= or USERHDRX= is specified.</td>
</tr>
</tbody>
</table>

**System action:** Processing continues.

**User response:** None. This message is informational.

---

HPSR0120I  SORT PROCESS IS NOT APPLICABLE (reason)

**Explanation:** SORT=COND is specified. However, physical sequential sorting is not processed by one of the following reasons:

<table>
<thead>
<tr>
<th>reason</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB ORGANIZATION</td>
<td>Database organization is not (P)HDAM.</td>
</tr>
<tr>
<td>USERHDR=</td>
<td>Both SORT= and USERHDR= are specified. These control statements are mutually exclusive when processing HALDBs.</td>
</tr>
<tr>
<td>USERHDRX=</td>
<td>Both SORT= and USERHDRX= are specified. These control statements are mutually exclusive when processing HALDBs.</td>
</tr>
</tbody>
</table>

**System action:** Processing continues.

**User response:** See the description of the SORT control statement and check the reason of the error.

---

HPSR0121W  PARTINIT=YES,UNLOAD IS IGNORED (reason)

**Explanation:** PARTINIT=YES,UNLOAD was ignored because IMS HP Load could not determine which partition should be initialized. *Reason* gives description why the process was ignored. The following list describes the *reason*:

- **CONCATENATED UNLOAD D/S**
  - The input unloaded data sets are concatenated.
- **LOAD API**
  - The Load API function is used.
- **LOW LEVEL UNLOAD D/S**
  - The input unloaded data set was not created by IMS HP Unload with the CHECKREC=YES option, or the release level or the maintenance level of IMS HP Unload is lower than the level that is required for IMS HP Load to work correctly.

**System action:** Processing continues.

**User response:** Check the HALDB Process Summary report to determine if the all deleted or defined partitions are processed. If there are partitions that are not processed, you must initialize them by DFSUPNT0 before using it.
HPSR0122W  PART: xxxxxxx IS NOT INITIALIZED
BECAUSE INFORMATION IS NOT
FOUND IN UNLOAD D/S
Explanation: The flag in the DBRC RECON record for
the partition shows that the partition needs to be
initialized (PINIT=ON). However, the partition whose
information is not found in the unloaded data set is not
initialized due to the specification of
PARTINIT=YES,UNLOAD.
System action: IMS HP Load continues processing
and sets a return code of 4.
User response: If the partition database data set is
deleted or defined before the reload, you must initialize
it by DFSUPNT0 before using it.

HPSR0123W  EMPTY PARTITION xxxxxxx IS NOT
INITIALIZED BECAUSE THE DBRC
FLAG SHOWS NOPINIT
Explanation: The unloaded partition does not contain
any segment during reload. However, the partition is
not initialized because the flag in the RECON shows
that the partition does not need to be initialized
(NOPINIT).
System action: IMS HP Load continues processing
and sets a return code of 4.
User response: If the partition database data set is
deleted or defined before the reload, you must initialize
it by DFSUPNT0 before using it.

HPSR0124E  WF1REC=NO CANNOT BE SPECIFIED
WITH IPR INDEXBLD=YES
Explanation: An incorrect control statement is
detected by the control statement syntax checking in
the Reload task of IPR Driver.
System action: Processing stops.
User response: Correct the error and rerun the job.

HPSR0125E  SORTHIDAM=YES IS NOT
APPLICABLE FOR db_type DB
Explanation: SORTHIDAM=YES is specified for the
Load API. However, the statement is not applicable for
the Load API.
System action: Processing stops.
User response: Correct the error and rerun the job.

HPSR0127E  USERHDR=, USERHDRX=, AND
SORTHIDAM=YES KEYWORDS ARE
MUTUALLY EXCLUSIVE
Explanation: All USERHDR, USERHDRX, and
SORTHIDAM=YES keywords are specified in the
FRRIN or the HPSIN data set. These keywords are
mutually exclusive when processing HALDBs.
System action: Processing stops.
User response: Correct the error and rerun the job.

HPSR0130W  EMPTY PART partition IS NOT
INITIALIZED BECAUSE NO
SPECIFICATION ON LPART
Explanation: In the reload process, IMS HP Load
found no segments in the unloaded data set for the
indicated partition. Although PARTINIT=YES,LPART
is specified, IMS HP Load does not initialize the empty
partition because the partition is not specified on the
LPART control statement.
System action: IMS HP Load continues processing
and sets a return code of 4.
User response: If the partition database data set is
deleted or defined before the reload, you must initialize
it by DFSUPNT0 before using it.

HPSR0131E  partname IS SPECIFIED
MORE THAN ONCE ON THE LPART
ST ATEMENT
Explanation: The indicated partition is specified more
than once on the LPART control statement. A partition
name that is already specified cannot be specified
again.
System action: Processing stops.
User response: Correct the parameter error, and rerun the job.

HPSR0132E UNDEFINED PARTITION NAME
(partname) IS SPECIFIED ON THE
LPART STATEMENT

Explanation: The indicated partition name is specified on the LPART control statement. However, the partition is not defined in the DBRC.

System action: Processing stops.

User response: Ensure that the indicated partition is specified correctly and that the partition is defined in the DBRC. Correct the error and rerun the job.

HPSR0133E THE EXIT NAME FOR USERHDRX=
CANNOT BE USED WITH SORT OR
SORTIDAM

Explanation: The exit routine name that is specified on the USERHDRX control statement cannot be specified with a SORT or a SORTIDAM control statement.

System action: Processing stops.

User response: See the list of exit routine names in “Restrictions for reloading user-defined header format records” on page 112, correct the exit routine name on the USERHDRX control statement, and rerun the job.

HPSR0134I LOGICAL RELATIONSHIP IS FOUND
IN THE DBD DEFINITION

Explanation: A logical relationship is defined in the DBD definition. If the database is a HALDB, you must rebuild the ILDS.

System action: IMS HP Load continues processing.

User response: None. This message is informational.

HPSR0135I SECONDARY INDEX IS DEFINED FOR
THE DB

Explanation: At least one secondary index is defined for the database. If the database is a HALDB, you must rebuild the ILDS or the secondary index databases.

System action: IMS HP Load continues processing.

User response: None. This message is informational.

HPSR0136E REPAIRILK=YES CANNOT BE
SPECIFIED FOR DB WITH LOGICAL
RELATIONSHIP

Explanation: REPAIRILK=YES cannot be specified for databases that are involved in logical relationships.

System action: Processing stops with return code 8.

User response: Correct the REPAIRILK control statement and rerun the job.

HPSR0137W ALTHOUGH REPAIRILK=YES IS
SPECIFIED, SOME PARTITIONS ARE
NOT RELOADED

Explanation: Although REPAIRILK=YES is specified, some partitions are not reloaded.

Attention: Segment ILKs of the reloaded partitions are reassigned, but segment ILKs of the unprocessed partitions remain the same. Under such a circumstance, duplicate segment ILKs might be produced within the database.

System action: IMS HP Load sets a return code of 4 and completes the reloading of the database.

User response: If you want to prevent duplicate segment ILKs, specify the REPAIRILK=YES option for all the partitions of the HALDB, and rerun the job.

HPSR0138W PSINDEX: ddbname MUST BE REBUILT

Explanation: The indicated PSINEX database must be rebuilt because ILKs are reassigned by the REPAIRILK=YES option.

System action: IMS HP Load continues processing.

User response: When the database is reorganized, rebuild the indicated PSINDEX database.

HPSR0201E xxxx TASK ENDED ABNORMALLY
(CC= comp_code)

Explanation: The subtask that is shown by string xxxx ended abnormally with a completion code of comp_code due to an error in the task. The subtask xxxx is as follows:

<table>
<thead>
<tr>
<th>xxxx</th>
<th>Subtask</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPL</td>
<td>Application task</td>
</tr>
<tr>
<td>DBRC</td>
<td>DBRC command task</td>
</tr>
<tr>
<td>ILES</td>
<td>ILE sort task</td>
</tr>
<tr>
<td>RELD</td>
<td>Reload task</td>
</tr>
<tr>
<td>SORT</td>
<td>Sort task of unloaded database data set</td>
</tr>
</tbody>
</table>

System action: One of messages HPSR3770E, HPSR4000E, or HPSR3775E follows, and IMS HP Load issues a user 3770, 4000, or 3775 abend.

User response: See the accompanying error messages that are issued from the abended task.
**HPSR0202E**  
**TASK WAS FORCED TO END DUE TO ERROR IN ANOTHER TASK**

Explanation: The task that is shown by string xxxx ended abnormally with U2006 due to an error in another task.

<table>
<thead>
<tr>
<th>xxxx</th>
<th>Subtask</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPL</td>
<td>Application task</td>
</tr>
<tr>
<td>RELD</td>
<td>Reload task</td>
</tr>
<tr>
<td>SORT</td>
<td>Sort task</td>
</tr>
</tbody>
</table>

System action: Message HPSR3770E or HPSR4000E follows, and IMS HP Load issues a user 3770 or 4000 abend.

User response: See the accompanying error messages that are issued from the abended task.

**HPSR0203I**  
**NO CALLS ARE PROCESSED BY LOAD API PROCESSING**

Explanation: LAPIPCBNO= is specified, but no calls are processed by the Load API processing.

System action: The processing continues, and all calls to the PCB are transferred to the DL/I call interface modules.

User response: If the database calls need to be processed by the Load API process, ensure that the specified PCB is correct. Otherwise, ignore this message and continue processing by using the DL/I module. Check the LAPIPCBNO, LAPIPDBD, and LAPIPCB control statements.

**HPSR0301W**  
**THE ACCESS TO OUTPUT REPOSITORY WAS CANCELED.**  
**REASON:** reason

Explanation: IMS HP Load canceled its access to the IMS Tools KB Output repository because initialization process failed. The list describes the reason:

<table>
<thead>
<tr>
<th>Reason</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITKBLOAD</td>
<td>Dynamic allocation for the library that is specified on the ITKBLOAD control failure.</td>
</tr>
<tr>
<td>DYNALLOC</td>
<td>Loading of the HKTXXLI module failed.</td>
</tr>
<tr>
<td>FAILURE</td>
<td>Opening of the library that is specified on the ITKBLOAD control statement failed.</td>
</tr>
<tr>
<td>LOAD</td>
<td>The library that is specified on the ITKBLOAD control statement is not APF-authorized.</td>
</tr>
</tbody>
</table>

System action: Processing continues, but none of the reports are stored in the IMS Tools KB Output repository.

User response: If you want to store reports in the IMS Tools KB Output repository, specify the IMS Tools KB product load module library correctly. If the reason is "ITKBLOAD NOT AUTHORIZED", the library must be APF-authorized.

**HPSR0312W**  
**ERROR OCCURRED IN ACCESSING OUTPUT REPOSITORY FUNC; function RC; return_code RSN; reason_code**

Explanation: An error occurred while getting access to the IMS Tools KB Output repository.

System action: Processing continues. If the return code is equal to or greater than 08, no reports are stored in the IMS Tools KB Output repository.

User response: If you want to store the reports in the IMS Tools KB Output repository, correct the error. If any of the messages, HPSR0320W, HPSR0321W, HPSR0322W, or HPSR0323W, which describe the cause of the error, are issued following this message, see the explanation for those messages. If these messages are not issued, check the return code and the reason code. The codes are in hexadecimal. For the description of the codes, see the Tools Base IMS Tools Knowledge Base User’s Guide or the Tools Base Configuration Guide.

**HPSR0302W**  
**OUTPUT REPOSITORY PROCESSING IS CANCELED**

Explanation: This informational message indicates that IMS HP Load canceled storing reports in the IMS Tools KB Output repository because Reload failed.

System action: Processing continues.

User response: None.

**HPSR0313W**  
**ERROR OCCURRED IN ACCESSING OUTPUT REPOSITORY FUNC; function RC; return_code RSN; reason_code**

Explanation: An error occurred while getting access to the IMS Tools KB Output repository.

System action: Processing continues. If the return
code is equal to or greater than 08, the report that is shown in the HPSR0314W message and the subsequent reports are not stored in the IMS Tools KB Output repository.

User response: See the explanation of the HPSR0314W message.

### HPSR0314W DBD: dbname PART: partition REPORT: report-name

Explanation: This message follows the HPSR0313W message to show which report is processed.

System action: Processing continues.

User response: See the explanation of the HPSR0313W message.

### HPSR0318W GETMAIN FAILED FOR ITKB TEMPORARY WORK STORAGE

Explanation: An error occurred in the GETMAIN macro when storing reports temporarily.

System action: Processing continues. Some of the reports are not stored in the IMS Tools KB Output repository.

User response: Ensure that sufficient virtual storage is available.

### HPSR0320W RECON ENTRY WAS NOT FOUND IN ITKB

Explanation: The RECON entry is not found in the IMS Tools Knowledge Base information management environment.

System action: Processing continues, but none of the reports are stored in the IMS Tools KB Output repository.

User response: If you want to store reports in the IMS Tools KB Output repository, register the RECON environment to your IMS Tools Knowledge Base information management environment. For more information about registering the RECON environment, see the Tools Base IMS Tools Knowledge Base User’s Guide or the Tools Base Configuration Guide.

### HPSR0321W ITKB SERVER NAME WAS INCORRECT

Explanation: The connection to the IMS Tools KB server failed because the server name specified by the ITKBSRVR control statement was incorrect.

System action: Processing continues, but none of the reports are stored to the IMS Tools KB Output repository.

User response: If you want to store reports in the IMS Tools KB Output repository, specify the correct IMS Tools KB server name.

### HPSR0322W THE PRODUCT IPR RELOAD WAS NOT DEFINED IN ITKB

Explanation: IMS HP Load is not defined in the IMS Tools Knowledge Base information management environment as a product that can store reports to the IMS Tools KB Output repository.

System action: Processing continues, but none of the reports that are produced by IMS HP Load are stored in the IMS Tools KB Output repository.

User response: If you want to store reports in the IMS Tools KB Output repository, register the IMS HP Load product by using the IMS Tools KB product administration utility. For more information, see the Tools Base IMS Tools Knowledge Base User’s Guide or the Tools Base Configuration Guide.

### HPSR0323W THE REPORT WAS NOT DEFINED IN ITKB

Explanation: The report is not defined in the IMS Tools Knowledge Base information management environment.

System action: Processing continues, but the report is not stored in the IMS Tools KB Output repository.

User response: If you want to store the report in the IMS Tools KB Output repository, register the report by using the IMS Tools KB product administration utility. For more information, see the Tools Base IMS Tools Knowledge Base User’s Guide or the Tools Base Configuration Guide.

### HPSR0400W ERROR CODE RETURNED FROM DB SENSOR. FUNC=function

Explanation: An error code is returned from the DB Sensor function. Function is the function name of the DB Sensor API.

System action: IMS HP Load stops calling the DB Sensor, continues processing, and sets a return code of 4.

User response: The accompanying messages are recorded in the CRSS journal. Check the CRSS journal and follow the user response sections of the accompanying messages.

### HPSR0401W INFORMATION REQUESTED BY DB SENSOR IS NOT VALID

Explanation: IMS HP Load received an invalid request from the DB Sensor API.

System action: IMS HP Load stops calling the DB Sensor, continues processing, and sets a return code of 4.

User response: This error is likely an internal error. Contact IBM Software Support.
HPSR2000I PIPE SERVICE INITIALIZATION SUCCESSFUL (IPRPIPE V.im.m.n)

**Explanation**: The IPRPIPE service is initialized successfully. The string V.im.m.n shows the internal version ID of IPRPIPE.

**System action**: The processing continues.

**User response**: None. This message is informational.

HPSR2000E PIPE SERVICE ['INIT' | 'TERM'] ERROR; REASON=xxxxxxxx

**Explanation**: An error occurred while initializing or terminating the IPRPIPE service. Reason code xxxxxxx shows the reason of the error.

**System action**: Program IPRPIPE returns a return code of 8.

**User response**: Contact IBM Software Support.

HPSR2001E PIPE SERVICE ERROR IN task OF PIPE ppppppppppppppp; REASON=reason_code

**Explanation**: A pipe service that is requested by the reader or the writer task, of pipe pppppppppppppp is aborted by an error in the task. The hexadecimal reason code reason_code shows the reason of the error and the status of the pipe at the time the error occurred. The reason code is the logical sum of the following reason flags:

<table>
<thead>
<tr>
<th>Flag (hex)</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>80000000</td>
<td>The pipe has been requested to open.</td>
</tr>
<tr>
<td>40000000</td>
<td>The pipe has been requested to close.</td>
</tr>
<tr>
<td>20000000</td>
<td>PUT processing for the pipe has been requested.</td>
</tr>
<tr>
<td>10000000</td>
<td>GET processing for the pipe has been requested.</td>
</tr>
<tr>
<td>01000000</td>
<td>Pipe OPEN, CLOSE, PUT, or GET processing is in progress.</td>
</tr>
<tr>
<td>00800000</td>
<td>The task has a lock.</td>
</tr>
<tr>
<td>00400000</td>
<td>The task is waiting for a lock.</td>
</tr>
<tr>
<td>00200000</td>
<td>The task is waiting for the partner task to put data in the pipe or to get data from the pipe.</td>
</tr>
<tr>
<td>00100000</td>
<td>This flag is used internally.</td>
</tr>
<tr>
<td>00010000</td>
<td>The pipe is closed on this end.</td>
</tr>
<tr>
<td>00008000</td>
<td>An error in the partner task is detected.</td>
</tr>
<tr>
<td>00004000</td>
<td>The task is ending abnormally.</td>
</tr>
<tr>
<td>00002000</td>
<td>A timeout occurred while a pipe request was being processed.</td>
</tr>
<tr>
<td>00001000</td>
<td>An error occurred while setting up a recovery routine.</td>
</tr>
<tr>
<td>00000800</td>
<td>A record that is passed to a PUT request is too large.</td>
</tr>
<tr>
<td>00000800</td>
<td>The partner task ended abnormally.</td>
</tr>
<tr>
<td>00000400</td>
<td>The partner task closed the pipe unexpectedly.</td>
</tr>
<tr>
<td>00000200</td>
<td>A timeout occurred while the task was waiting for the partner task to open the pipe.</td>
</tr>
</tbody>
</table>

**System action**: The task abends with a user completion code of 2005.

**User response**: Identify the cause of the error. If the cause is not found, contact IBM Software Support.

HPSR2002E ERROR IN ALLOCATING PIPE ppppppppppppppp; REASON=xxyyyyzz-ssssssss

**Explanation**: An error occurred while allocating pipe ppppppppppppppp. Reason code xxyyyyzz-ssssssss shows the reason of the error.

**System action**: Program IPRPIPE returns a return code of 8.

**User response**: Contact IBM Software Support.

HPSR2003E ERROR IN DEALLOCATING PIPE ppppppppppppppp; REASON=xxyyyyzz-aaaaaaa-bbbbbbbb

**Explanation**: An error occurred while deallocating pipe ppppppppppppppp. Reason code xxyyyyzz-aaaaaaa-bbbbbbbb shows the reason of the error.

**System action**: Program IPRPIPE returns a return code of 8.

**User response**: Contact IBM Software Support.

HPSR2004E TIME-OUT IN tttttt OF PIPE ppppppppppppppp; REASON=rrrrrrrr

**Explanation**: Timeout occurred while processing the writer or reader task tttttt through pipe ppppppppppppppp. Reason code rrrrrrrr shows the reason of the error.

**System action**: Message HPSR2006E follows and the task abends.

**User response**: Contact IBM Software Support.
PIPE SERVICE REQUESTED BY task IS ABORTED FOR PIPE

Explanation: A pipe service that is requested by the reader or the writer task task is aborted by an error in the task. The hexadecimal reason code reason_code shows the reason of the error and the status of the pipe at the time the error occurred. The reason code is the logical sum of the following reason flags:

Flag (hex) | Meaning
--- | ---
80000000 | The pipe has been requested to open.
40000000 | The pipe has been requested to close.
20000000 | PUT processing for the pipe has been requested.
10000000 | GET processing for the pipe has been requested.
01000000 | Pipe OPEN, CLOSE, PUT, or GET processing is in progress.
00800000 | The task has a lock.
00400000 | The task is waiting for a lock.
00200000 | The task is waiting for the partner task to put data in the pipe or to get data from the pipe.
00100000 | This flag is used internally.
00010000 | The pipe is closed on this end.
00008000 | An error in the partner task is detected.
00004000 | The task is ending abnormally.
00002000 | A timeout occurred while a pipe request was being processed.
00001000 | An error occurred while setting up a recovery routine.
00000800 | A record that is passed to a PUT request is too large.
00000400 | The task stopped waiting for the partner task to open the pipe.
00000080 | The partner task ended abnormally.
00000040 | The partner task closed the pipe unexpectedly.
00000020 | A timeout occurred while the task was waiting for the partner task to open the pipe.
00000010 | The partner task could not wait for the current task to open the pipe.
00000008 | The task was requested to stop waiting for the partner task to open the pipe.

User response: Identify the cause of the error in the partner task. If the cause is not found, contact IBM Software Support.

PIPE task ALREADY CLOSED THE PIPE

Explanation: The reader or the writer task, which is shown by task, tried to close the pipe, but the pipe was already closed.

User response: This error is likely caused by an internal error. Contact IBM Software Support.

PIPE task WAITING FOR PARTNER TO OPEN PIPE

Explanation: The reader or the writer task, which is shown by task, is waiting for the partner task to open the pipe.

User response: If this message is followed by message HPSR2009I that is issued from the partner task and message HPSR2010I, no action is required. If this message is followed by message HPSR2011E, something is wrong in the partner task. For example, the partner task abended before the pipe was opened.

PIPE task OPENED PIPE

Explanation: The reader or the writer task, which is shown by task, opened the pipe for which the partner task waited. Message HPSR2010I follows.

User response: None. This message is informational.

PIPE CONNECTION ESTABLISHED FOR PIPE

Explanation: The connection to pipe is established. This message is issued from the task that issued the preceding message HPSR2008I for the pipe.

User response: None. This message is informational.
<table>
<thead>
<tr>
<th>HPSR2011E</th>
<th>TIME-OUT WHILE PIPE</th>
<th>WAITING FOR PARTNER TO OPEN PIPE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explanation: If the partner task does not open the pipe within a minute, this message is issued from the task that issued the preceding message HPSR2008I for pipe. Something is wrong in the partner task.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>System action: Message HPSR2006E follows and the task abends.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User response: You can code the PIPEOWAIT control statement to stay in the wait state until the session of the pipe connection is established completely. For details of the statement, see “PIPEOWAIT control statement” on page 147. If the timeout still occurs even if the PIPEOWAIT control statement is coded, contact IBM Software Support.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>HPSR2998E</th>
<th>PIPE IS NOT OPENED. GPR2=ADDR(PDCB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explanation: The reader or the writer task tried to close the pipe that has not been opened.</td>
<td></td>
</tr>
<tr>
<td>System action: The task abends with a completion code of U2998.</td>
<td></td>
</tr>
<tr>
<td>User response: This error is likely an internal error. Contact IBM Software Support.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>HPSR3280E</th>
<th>HPSCLI00 IS NOT FOUND ALTHOUGH DECODEDBD=YES SPECIFIED</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explanation: Although DECODEDBD=YES is specified, the required module HPSCLI00 is not found.</td>
<td></td>
</tr>
<tr>
<td>System action: IMS HP Load issues a user 3380 abend and a dump.</td>
<td></td>
</tr>
<tr>
<td>User response: If you need to use the DECODEDBD=YES option, ensure that the required maintenance is applied to IMS Parallel Reorganization. Otherwise, remove the option and rerun the job.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>HPSR3601E</th>
<th>SPECIFIED DBD IS NOT FOUND IN DBD LIBRARY</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explanation: IMS HP Load could not find the DBD specified in the EXEC statement in the DBD library pointed to by the IMS DD statement.</td>
<td></td>
</tr>
<tr>
<td>System action: Module HPSRPDBD or HPSRHDBD issues a user 3601 abend.</td>
<td></td>
</tr>
<tr>
<td>User response: Ensure that the names of the DBD and the DBD library are specified correctly. If they are correct, the DBD might be missing and needs to be generated. Correct the error and rerun the job.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>HPSR3602I</th>
<th>HALDB dbdbname IS DEFINED AS OSAM8G</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explanation: The indicated HALDB is defined as OSAM8G in the RECON data set. The maximum capacity of OSAM data sets is 8 GB of data.</td>
<td></td>
</tr>
<tr>
<td>System action: IMS HP Load continues processing.</td>
<td></td>
</tr>
<tr>
<td>User response: None. This message is informational.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>HPSR3603E</th>
<th>POINTERS ACCOUNTING FAILURE IN A SEGMENT PREFIX</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explanation: IMS HP Load failed to account for all pointers in a segment prefix.</td>
<td></td>
</tr>
<tr>
<td>System action: Module HPSRPDBD or HPSRHDBD issues a user 3603 abend and a dump.</td>
<td></td>
</tr>
<tr>
<td>User response: Contact IBM Software Support.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>HPSR3607E</th>
<th>UNSUPPORTED DATABASE TYPE SPECIFIED</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explanation: The specified DBD shows that the database is not HDAM/PHDAM, HIDAM/PHIDAM, or HISAM/SHISAM. These organizations are the only database organizations that IMS HP Load can process.</td>
<td></td>
</tr>
<tr>
<td>System action: Module HPSRPDBD or HPSRHDBD issued a user 3607 abend.</td>
<td></td>
</tr>
<tr>
<td>User response: Correct the error and rerun the job.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>HPSR3608E</th>
<th>LCHILD FOR HIDAM PRIMARY INDEX IS NOT DEFINED IN DBD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explanation: IMS HP Load was unable to find an LCHILD definition for the HIDAM primary index DBD.</td>
<td></td>
</tr>
<tr>
<td>System action: Module HPSRPDBD issues a user 3608 abend and a dump.</td>
<td></td>
</tr>
<tr>
<td>User response: Check the HIDAM database DBD to add or to change the LCHILD statement that defines the primary index. Correct the error and rerun the job.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>HPSR3609E</th>
<th>UNABLE TO LOCATE A VALID DBD FOR HIDAM PRIMARY INDEX</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explanation: IMS HP Load was unable to locate a valid DBD for the HIDAM primary index.</td>
<td></td>
</tr>
<tr>
<td>System action: Module HPSRPDBD issues a user 3609 abend and a dump.</td>
<td></td>
</tr>
<tr>
<td>User response: Ensure that the correct DBD library is specified on the IMS DD statement. If so, add the correct DBD to the library. If not, correct the DD statement. Rerun the job.</td>
<td></td>
</tr>
</tbody>
</table>
**HPSR3610E SYNAD EXIT WAS INVOKED FOR A SEQUENTIAL WRITE**

**Explanation:** The SYNAD exit was invoked for a sequential WRITE.

**System action:** Module HPSROSAM issues a user 3610 abend and a dump.

**User response:** Register 3 points to the address of a BSAM DECB, and Register 2 points to the address of the data set DCB. Check the DECB exception bits for the error as described in *DFSMS Using Data Sets*.

If the problem persists, collect the dump in SYSUDUMP and contact IBM Software Support.

**HPSR3611E SYNAD EXIT WAS INVOKED FOR AN UPDATE PROCESS**

**Explanation:** The SYNAD exit was invoked for an update process.

**System action:** Module HPSROSAM issues a user 3611 abend and a dump.

**User response:** Register 3 points to the address of a BSAM DECB, and Register 2 points to the address of the data set DCB. Check the DECB exception bits for the error as described in *DFSMS Using Data Sets*.

If the problem persists, collect the dump in SYSUDUMP and contact IBM Software Support.

**HPSR3612E SYNAD EXIT WAS INVOKED FOR A VSAM DATA SET**

**Explanation:** The SYNAD exit was invoked for a VSAM data set.

**System action:** Module HPSRVSAM issues a user 3612 abend and a dump.

**User response:** Register 3 contains the feedback codes from the RPL. Register 5 contains the address of the ACB. Look up the error reason flag in field ACBERFLG. The error codes are explained in the *DFSMS/MVS Macro Instructions for Data Sets*.

Correct the error and rerun the job.

**HPSR3614E I/O FAILURE OCCURRED DURING A VSAM SEQUENTIAL WRITE**

**Explanation:** An error occurred attempting a VSAM sequential write.

**System action:** Module HPSRVSAM issues a user 3614 abend and a dump.

**User response:** Register 3 contains the RPL feedback code, and register 5 contains the RPL address. The second byte of register 3 contains the return code, and the fourth byte contains the reason code. The explanation of these codes is in the *DFSMS/MVS Macro Instructions for Data Sets*.

Correct the error and rerun the job.

**HPSR3616E I/O FAILURE OCCURRED DURING A SEQUENTIAL VSAM PUT**

**Explanation:** A sequential VSAM PUT was not accepted because the RPL was active for another request.

**System action:** Module HPSRVSAM issues a user 3616 abend and a dump.

**User response:** Contact IBM Software Support.

**HPSR3617E I/O FAILURE OCCURRED DURING A DIRECT VSAM PUT**

**Explanation:** A direct VSAM PUT was not accepted.

**System action:** Module HPSRVSAM issues a user 3617 abend and a dump.

**User response:** Contact IBM Software Support.

**HPSR3618E VSAM ESDS OPEN FAILURE OCCURRED**

**Explanation:** An error occurred while a VSAM ESDS data set was being opened.

**System action:** Module HPSRVSAM issues a user 3618 abend and a dump.

**User response:** Register 5 contains the address of the ACB. Look up the error reason flag in field ACBERFLG. The error codes are explained in the *DFSMS/MVS Macro Instructions for Data Sets*.

Correct the error and rerun the job.

**HPSR3619E DD NOT FOUND FOR DATA SET GROUP “ddddd”**

**Explanation:** No DD statement for data set group “ddddd” is found in the JCL. The DD statement for the data set group is required.

**System action:** IMS HP Load issues a user 3619 abend and a dump.

**User response:** Check if the DD statement for “ddddd” exists in the JCL. Correct the error and rerun the job.

**HPSR3620E HISAM/SHISAM PRIMARY DATA SET (KSDS) OPEN FAILURE OCCURRED**

**Explanation:** An error occurred while a VSAM KSDS data set was being opened.

**System action:** IMS HP Load issues a user 3620 abend and a dump.
User response: Register 5 contains the address of the ACB. Look up the error reason flag in field ACBERFLG.
Correct the error and rerun the job.

HPSR3621E INCORRECT DATA SET FOR ddbname DD - reason

Explanation: Incorrect parameter is specified in ddbname DD, reason shows the error reason.

<table>
<thead>
<tr>
<th>Reason</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DUMMY</td>
<td>A dummy data set is specified for ddbname.</td>
</tr>
<tr>
<td>DISP=MOD</td>
<td>DISP=MOD is specified for ddbname.</td>
</tr>
</tbody>
</table>

System action: IMS HP Load issues a user 3621 abend and a dump.
User response: Ensure that a DD statement is present for the ddbname indicated, and that it properly identifies the correct data set. Correct the error and rerun the job.

HPSR3622E DD NOT FOUND HISAM OVERFLOW DATA SET (ESDS) "dddddddd"

Explanation: No DD statement for the HISAM overflow data set "dddddddd" is found in the JCL. Such a DD statement is required.
System action: IMS HP Load issues a user 3622 abend and a dump.
User response: Check whether the DD statement for "dddddddd" exists in the JCL. Correct the error and rerun the job.

HPSR3623E TRKCALC FAILED FOR ddbname, RC= return_code

Explanation: IMS HP Load called TRKCALC, but received an error return code from TRKCALC. Return_code shows the return code.
System action: IMS HP Load issues a user 3623 abend and a dump.
User response: See the DFSMSdfp Advanced Services and check the meaning of the return code of TRKCALC. Correct the error and rerun the job.

HPSR3624E I/O FAILURE OCCURRED DURING A VSAM DIRECT READ

Explanation: An error occurred on a VSAM direct read.
System action: Module HPSRVSAM issues a user 3624 abend and a dump.
User response: Register 3 contains the RPL feedback code, and register 5 contains the address of the RPL. The second byte of register 3 contains the request return code, and the fourth byte contains the reason code. The meaning of these codes is in the DFSMS/MVS Macro Instructions for Data Sets.
Correct the error and rerun the job.

HPSR3625E I/O FAILURE OCCURRED DURING A VSAM DIRECT WRITE

Explanation: An error occurred on a VSAM direct write.
System action: Module HPSRVSAM issues a user 3625 abend and a dump.
User response: Register 3 contains the feedback codes from the RPL. Register 5 contains the address of the RPL. The second byte of register 3 contains the return code, and the fourth byte contains the reason code. The explanation of these codes is in the DFSMS/MVS Macro Instructions for Data Sets.
Correct the error and rerun the job.

HPSR3626E proc_name ERROR FOR DDNAME: ddbname DURING ESTAE PROCESSING

Explanation: The process (proc_name) for the ddbname data set ended with an error in ESTAE processing.
System action: IMS HP Load ends the process.
User response: See the accompanying error messages.

HPSR3628E DYNAMIC VOLUME ADDITION IS NOT ALLOWED

Explanation: While processing an OSAM database, IMS HP Load found a volume that is dynamically added to the volume list after the processing started.
System action: Module HPSROSAM issues a user 3628 abend and a dump.
User response: If the database to be reloaded requires multiple DASD volumes, define explicitly enough multi-volumes for the database.

HPSR3629I ESTAE RECOVERY ROUTINE IS INVOKED

Explanation: The ESTAE routine is invoked to close the database data sets.
System action: IMS HP Load ends the process.
User response: See the accompanying error messages.
**HPSR3630E**  
RAP ASSIGNED OUT OF PHYSICAL SEQUENCE

**Explanation:** Root key data was not in the order of hexadecimal value `bbbbbbrr` that represents the result of randomizing module conversion. `bbbbbb` is the block number and `rr` is the RAP number.

**System action:** Module HPSRHDL0 or HPSRRDLM issues a user 3630 abend.

**User response:** If the randomizer was changed, specify 'YES' for the SOR control statement in FRRIN and run Load to sort the unloaded data in the order `bbbbbbrr`. If you use PSSR and the non-zero byte limit value is specified for your database, you must specify 'NO' in column 51 in the PSSR control statement and run PSSR before you rerun Load to sort. To avoid a degradation of performance in reloading, it is recommended that you specify `-nnn` on the SEARCH control statement.

---

**HPSR3631E**  
A SEQUENCE ERROR DETECTED ON DF SUINPT

**Explanation:** There is an error in the segment sequence on the HD unloaded data set (DFSUINPT DD).

**System action:** Module HPSRPTRS issues a user 3631 abend and a dump.

**User response:** Probable cause can be: the HD unloaded data set is a multivolume file, or tapes are not mounted in the appropriate sequence.

Rerun the job, ensuring that the volumes are mounted in the correct sequence.

---

**HPSR3632E**  
A SEQUENCE FIELD IS LESS THAN THE PREVIOUS SEGMENT

**Explanation:** IMS HP Load attempted to put a segment chain whose sequence field is less than the previous segment on a twin chain.

**System action:** Module HPSRPTRS issues a user 3632 abend and a dump.

**User response:** Check the original database, and correct the incorrect sequence field. Rerun the job.

---

**HPSR3633E**  
A SEQUENCE FIELD IS NOT UNIQUE

**Explanation:** IMS HP Load attempted to put a segment whose sequence field was not unique on a twin chain. But the specification of the sequence field in the DBD `dbdname` is unique. This message is followed by the key data.

The DBD name, segment name, sequence field name, and key data are shown in the Messages report.

**System action:** Module HPSRPTRS issues a user 3633 abend and a dump.

**User response:** Check the sequence field for this segment in the DBD. Ensure that it is marked unique. It should be changed to multiple, or locate and correct the sequence field in the original database. Correct the error and rerun the job.

---

**HPSR3634E**  
A ROOT SEGMENT IS OUT OF SEQUENCE

**Explanation:** IMS HP Load found a HIDAM/HISAM/SHISAM/PHIDAM root segment whose sequence field was out of sequence. This message is followed by the key data.

The DBD name, segment name, sequence field name, and key data are shown in the Messages report.

**System action:** Module HPSRPTRS issues a user 3634 abend and a dump.

**User response:** Locate the segment description in the DBD, and add the PTF pointer if needed. Correct the error and rerun the job.

---

**HPSR3635E**  
TWIN-FORWARD CHAIN IS NOT SPECIFIED FOR HDAM/PHDAM SYNONYM

**Explanation:** A synonym of an HDAM/PHDAM root segment is found, but no physical twin forward (PTF) in the root segment is specified for the synonym; that is, POINTER=NOTWIN is coded in the DBD.

**System action:** Module HPSRPTRS issues a user 3635 abend and a dump.

**User response:** Locate the segment description in the DBD, and add the PTF pointer if needed. Correct the error and rerun the job.

---

**HPSR3636E**  
THE INCONSISTENCY OF DEPENDENT SEGMENT WAS DETECTED

**Explanation:** A second dependent segment has occurred, but that segment type does not have a physical twin forward (PTF) or hierarchical forward (HF) pointer defined.

**System action:** Module HPSRPTRS issues a user 3636 abend and a dump.

**User response:** Check the DBD to see if the segment in error has a PTF or HF pointer. Correct the error and rerun the job.
HPSR3637E  NUMBER OF KEY SEQUENCE ERRORS EXCEEDS THE LIMIT: nnnnn

Explanation: The number of key sequence errors exceeds the specified limit value. nnnnn is a value that is specified by SEQERROR=SAVE, nnnnn, or SEQERROR=ACCEPT, nnnnn.

System action: IMS HP Load issues a user 3637 abend.

User response: If necessary, check the original database and the incorrect sequence field.

If you want to allow any number of sequence errors, specify SEQERROR=SAVE or SEQERROR=ACCEPT without the limit value nnnnn. Otherwise, correct the sequence errors in the unloaded data set, and rerun the job.

HPSR3638E  INTERNAL ERROR OCCURRED DURING POINTER RESOLUTION

Explanation: An unexpected internal error occurred during the pointer resolution process.

System action: IMS HP Load issues a user 3638 abend and a dump.

User response: Contact IBM Software Support.

HPSR3639E  NON-HIERARCHICAL SEQUENCE SEGMENT WAS DETECTED IN DFSUINPT

Explanation: The HD unloaded data set (DFSUINPT DD) contains segments that are not in hierarchical sequence.

System action: Module HPSRPTRS issues a user 3639 abend and a dump.

User response: Check the results of HD unload processing. Correct the error and rerun the job.

HPSR3640E  DYNALLOC FAILURE OCCURRED FOR xxxxxxx: RETURN CODE=return_code REASON CODE=reason_code

Explanation: An attempt of dynamic allocation for the indicated data set failed. return_code and reason_code are the return code and the reason code from the SVC99 routine.

System action: A user 3640 abend and a dump are issued.

User response: See the z/OS MVS Programming: Authorized Assembler Services Guide for the return code and the reason code from the SVC99 routine. Correct the error and rerun the job.

HPSR3641E  data_set OPEN FAILURE OCCURRED

Explanation: An attempt to open the data set data_set failed.

System action: A user 3641 abend and a dump are issued.

User response: Check the system messages in the job log. Correct the error and rerun the job.

HPSR3642E  I/O FAILURE OCCURRED DURING xxxx FOR yyyyyyyyy

Explanation: An attempt for input/output to the data set yyyyyyyyy failed. xxxx is the type of I/O, READ or WRITE.

System action: A user 3642 abend and a dump are issued.

User response: Check the system messages in the job log. Correct the error and rerun the job.

HPSR3643E  DSPSER xxxxxx FAILURE OCCURRED: RETURN CODE is yyyy REASON CODE is zzzzzzzzz

Explanation: The DSPSERV macro failed. xxxxxx is the function code of the failed macro, yyyy is the return code, and zzzzzzzzz is the reason code.

System action: A user 3643 abend and a dump are issued.

User response: See the z/OS MVS Programming: Assembler Services Reference for the return code and the reason code from DSPSERV macro.

Correct the error, and rerun the job. If your installation limits the amount of data space that the IEFUSI exit routine can use, consult your system programmer to extend the allocation.

HPSR3644E  ALESERV xxxxxx FAILURE OCCURRED: RETURN CODE is yyyy

Explanation: The ALESERV macro failed. xxxxxx is the function code of the failed macro, and yyyy is the return code.

System action: A user 3644 abend and a dump are issued.

User response: See the z/OS MVS Programming: Assembler Services Reference for the return code and the reason code from the ALESERV macro. Correct the error, and rerun the job.

HPSR3646E  SORT CANNOT BE PROCESSED DUE TO COMPRESSED ROOT KEY

Explanation: IMS HP Load cannot sort the unload data, of which root key field is compressed.
System action: IMS HP Load issues a user 3646 abend and a dump.

User response: Specify DECOMPRESS=YES for the IMS HP Unload utility job to expand the compressed keys.

HPSR3647E  THE SIZE OF SORTWK CANNOT BE ESTIMATED, BECAUSE THE BLKSIZE OF DBDS IS ZERO.

Explanation: The size of intermediate storage data sets used by DFSORT cannot be estimated because the BLKSIZE of the DBDS is zero.

System action: IMS HP Load issues a user 3647 abend and a dump.

User response: Correct the error and rerun the job.

HPSR3648E  HPSRSVX2 EXIT CANNOT RUN WITH SORTHIDAM=YES

Explanation: The HPSRSVX2 exit cannot run with SORTHIDAM=YES.

System action: IMS HP Load issues a user 3648 abend and a dump.

User response: Correct the error and rerun the job.

HPSR3649I  AVERAGE SEGMENT LENGTH FOR SORTWK ESTIMATION IS ROUNDED UP

Explanation: IMS HP Load calculated the average segment size to estimate the size of sort work data sets. The calculated value is rounded up.

System action: IMS HP Load continues processing.

User response: None. This message is informational.

HPSR3650E  QSAM PUT ERROR OCCURRED FOR WFP DATASET: data_set

Explanation: A QSAM put error occurred while processing the indicated WFP data set.

System action: IMS HP Load issues a user 3650 abend and a dump.

User response: Register 3 points to the address of the status indicators. See DFSMSdfp Diagnosis to interpret the exception from status indicators. Correct the error and rerun the job.

HPSR3651E  SEGMENT WAS LOADED BEYOND THE 4GB SIZE LIMIT

Explanation: IMS HP Load utility reloaded a segment that is beyond the 4- (or 8-) GB size limit.

System action: Module HPSRHDLD or HPSRRLDM issues a user 3651 abend.

User response: If you are reloading an unloaded data set that was created from an OSAM database whose size is over 4 GB, check that the block size of the output database data set is an even number and that the database is a non-HALDB.

If the block size is correct, the database data set is full. If you are reloading an HDAM/PHDAM database, adjust the SEARCH control statement value to decrease the number of segments directed to overflow portion of the database. If so, rerun the job. If this situation persists, plan to divide the database data set into two or more data set groups. Correct the error and rerun the job.

HPSR3652E  RAA GREATER THAN THE 4GB SIZE LIMIT

Explanation: For the HDAM/PHDAM database, IMS HP Load utility detected that RAA exceeded the 4- (or 8-) GB size limit, and RAAFORMAT=NO was not specified.

System action: Module HPSRHDLD or HPSRRLDM issues a user 3652 abend.

User response: Specify RAAFORMAT=YES for reloading the HDAM/PHDAM database whose RAA size exceeded the limit.

HPSR3659E  DDN dd_name IS NOT FOUND IN THE RECON

Explanation: IMS HP Load checks the block size of each database data set with the size that is registered to RECON. However, the DDN dd_name is not found in the RECON data set.

System action: IMS HP Load issues a user 3659 abend and a dump.

User response: Ensure that the indicated DD name is correct. If necessary, contact IBM Software Support.

HPSR3660E  THE BLKSIZE OF DDN=dd_name DOES NOT MATCH THE VALUE IN RECON

Explanation: The block size of the database data set is not the same as the block size that is registered to RECON.

System action: IMS HP Load issues a user 3660 abend and a dump.

User response: If you specified the block size for the database data set in the JCL, remove the block size specification or correct the value, and then rerun the job. Alternatively, you can change the RECON definition from NOREORGV to REORGV so that the correct block size is reloaded. If necessary, contact IBM Software Support.
HPSR3667E  INPUT FORMAT *CS IS NOT ALLOWED FOR SHISAM

Explanation: The input format CS (the communication industry standard format), which is not allowed, is used for SHISAM.

System action: IMS HP Load issues a user 3667 abend and a dump.

User response: If the database to be reloaded is a SHISAM database, check the format of the unloaded data set and the DBD. If you want to reload a SHISAM, you must use the HD or SH format as input. Use one of these formats as input, and rerun the job.

HPSR3668E  INPUT FORMAT IS NOT ALLOWED FOR NON-HALDB

Explanation: The input format PHD or CP. They are not supported for a non-High Availability Large Database.

System action: IMS HP Load issues a user 3668 abend and a dump.

User response: If the database to be reloaded is not a HALDB, check the format of the unloaded data set and the DBD. If you want to reload a non-HALDB, you must use the HD, SH, or the CS format as input. Use one of these formats as input, and rerun the job.

HPSR3669E  INPUT FORMAT IS NOT ALLOWED FOR HALDB

Explanation: The input format HD or SH (the short header format), which is not allowed, is used for a High Availability Large Database.

System action: IMS HP Load issues a user 3669 abend and a dump.

User response: If you want to reload a High Availability Large Database, the PHD format as input, and rerun the job. If the database to be reloaded is not a HALDB, check the format of the unloaded data set and the DBD.

HPSR3670E  INPUT FORMAT *SH/*CS/*CP/ HDX IS NOT ALLOWED FOR A DATABASE IN A LOGICAL RELATIONSHIP

Explanation: The input format SH (the short header format), CS (the communication industry standard format), CP (the communication industry partitioned format), or HDX (USERHDRX or USERHDR control statement is specified) is used for a database in a logical relationship, which is not supported.

System action: IMS HP Load issues a user 3670 abend and a dump.

User response: Check the unloaded data set format and the DBD if the database to be reloaded has a logical relationship. If you want to reload a HALDB that has logical relationships, specify the unloaded file that is in the HD format as input, and rerun the job.

HPSR3671E  INPUT FORMAT IS INCORRECT FOR A DATABASE IN A LOGICAL RELATIONSHIP

Explanation: The format of the input unloaded data set is not applicable for a HALDB with logical relationships.

System action: IMS HP Load issues a user 3671 abend and a dump.

User response: Correct the error and rerun the job.

HPSR3672E  UNEXPECTED PARTITION CHANGE, REASON=xx

Explanation: This message is issued when IMS HP Load is running as the Reload task of the IPR Driver. An unexpected partition change occurred while the unloaded segment records were passed from the Unload task to the Reload task. The reason for the error is indicated by code xx:

<table>
<thead>
<tr>
<th>Reason (xx)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>The Unload task issued a notice of the partition change before the Reload task received the header record.</td>
</tr>
<tr>
<td>02</td>
<td>Partition changed or ended without notification.</td>
</tr>
</tbody>
</table>

System action: The Reload task issues a user abend 3672.

User response: Correct the error and rerun the job. This error is likely an internal system error. If necessary, contact IBM Software Support.

HPSR3673E  A DIFFERENT PARTITION IS ASSIGNED BY A ROOT KEY CHANGE IN USER EXIT ROUTINE

Explanation: The exit routine that is used in the Reload task of the IPR Driver job modified the sequence key value of a root segment, and the partition assigned to the segment was changed from the original one. When IMS HP Load is running as the Reload task of the IPR Driver, any exit routine for IMS HP Load must not modify a root sequence key to change the partition assigned to the segment.
System action: The Reload task issues a user abend 3673.

User response: Correct the error and rerun the job.

**HPSR3674E**  DBDNAME dbdname IN CS HEADER DOES NOT MATCH DBD SPECIFIED

Explanation: The DBD name in CS header must match the DBD name that is specified in the EXEC parameters.

System action: The Reload task issues a user abend 3674.

User response: Correct the DBDNAME in the CS header or the EXEC parameters, and rerun the job.

**HPSR3675E**  UNABLE TO PROCESS THE UNLOAD FILE DUE TO INCORRECT STAT RECORD

Explanation: The statistics records in the Unload data set are not recognizable by the Reload utility.

System action: The Reload task issues a user 3675 abend.

User response: Contact IBM Software Support.

**HPSR3676E**  THE INPUT FORMAT IS NOT ALLOWED FOR A HALDB WITH A PSINDEX

Explanation: The input format CS (the communication industry standard format) is used for a HALDB with a partitioned secondary index, which cannot be used.

System action: IMS HP Load issues a user 3676 abend and a dump.

User response: If the database to be reloaded is HALDB with a partitioned secondary index, check the unloaded data set format and the DBD.

If you want to reload a HALDB with a partitioned secondary index, use the PHD format as input, and rerun the job.

**HPSR3678E**  THE RECORD SIZE IS NOT LARGE ENOUGH TO CONTAIN THE LONGEST SEGMENT. DDN:ddname

Explanation: The record size for the database data set is not large enough to contain the longest segment in the database plus the IMS overhead required.

System action: IMS HP Load issues a user 3678 abend and a dump.

User response: Correct the record size of the database data set. The recommended record size is provided in the output listing of DBD generation.

**HPSR3679E**  OVERFLOW RECORD SIZE CANNOT BE LESS THAN PRIME RECORD SIZE.

Explanation: Logical record lengths in the overflow data set must be equal to or greater than the logical record length in the primary data set.

System action: IMS HP Load issues a user 3679 abend and a dump.

User response: Correct the logical record lengths of database data sets.

**HPSR3683I**  CHECKING DBDS USAGE IS BYPASSED DUE TO ['GETDSAB'|'SWAREQ'] FAILURE FOR DDNAME:ddname

Explanation: Due to a failure in issuing the GETDSAB or the SWAREQ macro, IMS HP Load could not obtain the data set name of the database data set. IMS HP Load did not check whether the database data set is being used or not.

System action: IMS HP Load continues processing.

User response: If you want IMS HP Load to check whether the database data set is in use or not, contact IBM Software Support.

**HPSR3684I**  CHECKING DBDS USAGE IS BYPASSED DUE TO GQSCAN ERROR RC=$, RSN=$

Explanation: Because IMS HP Load received an error return code and reason code as result of issuing the GQSCAN macro, IMS HP Load did not check whether the database data set is being used or not.

System action: IMS HP Load continues processing.

User response: If you want IMS HP Load to check whether the database data set is in use or not, contact IBM Software Support.

**HPSR3685E**  THE FOLLOWING DBDS IS IN USE BY JOBNAME=job_name ON SYSTEM=sysname1, sysname2 DSN=data_set_name

Explanation: The indicated database data set is used by another job (job_name). If the database data set is used by a job in another system and the name of that job is the same as the IMS HP Load job name, sysname1 and sysname2 are printed to identify the system.

System action: IMS HP Load issues a user 3685 abend.

User response: Rerun the IMS HP Load job after the other job releases the database data set.
HPSR3686W  DSN=data_set_name IS NOT USED.

Explanation: The indicated database data set is not used.

System action: IMS HP Load continues processing.

User response: None.

HPSR3687E  ERROR RETURNED FROM GQSCAN MACRO RC=$, RSN=$

Explanation: The GQSCAN macro failed.

System action: IMS HP Load issues a user 3687 abend.

User response: Contact IBM Software Support.

HPSR3688I  CHECKING DBDS USAGE IS BYPASSED DUE TO THE LACK OF AUTHORIZATION

Explanation: Multilevel security is activated and the program is not authorized to issue the GQSCAN macro. IMS HP Load did not check whether the database data set is being used or not.

System action: IMS HP Load continues processing.

User response: If you want IMS HP Load to check whether the database data set is in use or not, ensure that the program is running authorized, or is associated with your userid with at least READ access to the best fit FACILITY class resource profile of the form ISG.QSCANSERVICES.AUTHORIZATION.

HPSR3689E  DATABASE AUTHORIZATION FAILURE OCCURRED

Explanation: IMS HP Load received a bad return code from the IMS authorization module DFSDBAU0.

System action: Module HPSRHDLD issues a user 3693 abend.

User response: Ensure that the database is authorized for reload. Correct the error and rerun the job.

HPSR3690E  DBRC REQUEST FAILED, REASON=reason_code, RC=return_code

Explanation: The DBRC request with the reason code reason_code failed with a return code of return_code. To determine the cause of the failure that is specified by the reason code, see DFS030I in IMS Messages and Codes. And to determine the cause of the failure that is specified by the return code and for information about DBRC requests and return codes, also see IMS Messages and Codes.

System action: IMS HP Load issues a user 3690 abend.

User response: Correct the error and rerun the job.

HPSR3691E  A SEGMENT NAME IN DFSUINPT IS NOT FOUND IN DBD

Explanation: IMS HP Load is unable to find a segment name in the DBD that matches the one from the HD Unload data set (DFSUINPT DD).

System action: Module HPSRHDLD or HPSRRLDM issues a user 3691 abend and a dump.

User response: Register 9 points to the input record. The segment name is 6 bytes into the record. Check to be certain you are running with the correct DBD.

Correct the error and rerun the job.

HPSR3692E  HIDAM/PHIDAM INDEX (KSDS) OPEN FAILURE OCCURRED

Explanation: IMS HP Load unsuccessfully attempted to OPEN a VSAM KSDS for the HIDAM/PHIDAM Primary Index.

System action: Module HPSRHDLD or HPSRRLDM issues a user 3692 abend.

User response: Register 5 contains the address of the ACB. Examine the ACB error field (ACBERFLG) to determine the reason.

Correct the error and rerun the job.

HPSR3693E  HIDAM/PHIDAM INDEX (KSDS) I/O FAILURE OCCURRED

Explanation: IMS HP Load has had an unsuccessful PUT to the VSAM KSDS for the HIDAM/PHIDAM Primary Index.

System action: Module HPSRHDLD or HPSRRLDM issues a user 3693 abend.

User response: Register 5 contains the address of the RPL, and register 3 contains the RPL feedback code. Examine the second byte of register 3 to locate the return code and the fourth byte for the reason code.

Correct the error and rerun the job.

HPSR3694E  KEYLEN VALUE OF VSAM CATALOG IS NOT EQUAL TO THE DBD ONE

Explanation: The KEYLEN field in the VSAM catalog is not the same as the corresponding value in the DBD for the HIDAM/PHIDAM primary index or HISAM/SHISAM primary data set.

System action: Module HPSRHDLD or HPSRRLDM issues a user 3694 abend.

User response: Identify the incorrect value, correct the error, and rerun the job.
HPSR3695E VSAM KSDS WITH EXTENDED ADDRESSABILITY ATTRIBUTE IS NOT SUPPORTED

Explanation: The KSDS is an SMS data set with the extended addressability attribute, which IMS does not support.

System action: IMS HP Load issues a user 3695 abend and a dump.

User response: Allocate the data set with a data class that does not specify extended addressability, and rerun the job.

HPSR3696E MAXIMUM RELATIVE BLOCK NUMBER VALUE IS NOT SPECIFIED

Explanation: The DBD does not specify a maximum relative block number value for an HDAM/PHDAM database. IMS HP Load requires a value because it preformats the entire root addressable area.

System action: Module HPSRHDLD or HPSRLDM issues a user 3696 abend.

User response: Use the IMS HD Reorganization Reload utility (DFSURGL0) to reorganize a database with an "infinite" root addressable area.

HPSR3697E GETMAIN FAILED

Explanation: A GETMAIN failed while attempting to get storage for a bitmap block.

System action: IMS HP Load issues a user 3697 abend and a dump.

User response: Determine how much more storage must be allocated for the job step. Increase the job-step region size, and rerun the job.

HPSR3698E DFSUINPT OPEN FAILURE OCCURRED

Explanation: IMS HP Load was unsuccessful in attempting to OPEN the HD unloaded data set (DFSUINPT DD).

System action: Module HPSRHDLD or HPSRLDM issues a user 3698 abend.

User response: Ensure that a DFSUINPT DD statement exists in the JCL. If there is, check the JCL system messages to find the reason for the unsuccessful OPEN.

Correct the error and rerun the job.

HPSR3701E RELOAD TERMINATED - TOO MANY SEGMENTS DROPPED

Explanation: IMS HP Load received a return code of 4 from the HDAM randomizer too many times, and more than 10,000 segments (root segments and their dependent segments) were dropped.

System action: Module HPSRHDLD or HPSRLDM issues a user 3701 abend.

User response: Check the reason why the randomizer issued a return code 4 so many times. Correct the error and rerun the job.

HPSR3702E RETURN CODE xx FROM HDAM RANDOMIZER

Explanation: IMS HP Load received a return code xx (in hexadecimal) from the HDAM randomizer.

System action: Module HPSRHDLD or HPSRLDM issues a user 3702 abend.

User response: Check the reason of the return code xx from the HDAM randomizer. Correct the error and rerun the job.

HPSR3703E UNSUPPORTED LEVEL OF IMS IS BEING USED: level IMS LEVEL OF THIS RUN

Explanation: You are running an IMS batch region controller (DFSRRC00) that is not supported by IMS HP Load. level represents IMS level.

System action: IMS HP Load issues a user 3703 abend.

User response: Run IMS HP Load with the correct version of IMS.
REQUESTED USER EXIT ROUTINE "routine" NOT FOUND

Explanation: IMS HP Load attempted to load the user exit routine "routine", but the load module is not found.

System action: IMS HP Load issues a user 3704 abend and a dump.

User response: Ensure that the correct user exit program library is concatenated to the STEPLIB DD.

TERMINATION REQUESTED BY USER EXIT ROUTINE, RETURN CODE=return_code

Explanation: Job step termination is requested by the user exit routine; returning a return code of return_code.

System action: The reload job ends immediately with the indicated return code. The subsequent unloaded segments are not reloaded.

User response: Check the return code of the user exit routine. If the return code is not intended, correct the error and rerun the job.

INVALID RETURN CODE FROM USER EXIT ROUTINE, RETURN CODE=return_code RETURN CODE GREATER THAN 999

Explanation: An incorrect return code return_code or a value that is greater than 999 is returned by the user exit routine.

System action: IMS HP Load issues a user 3706 abend and a dump.

User response: Correct the error and rerun the job.

INCORRECT REQUEST FROM USER EXIT ROUTINE, RETURN CODE=return_code reason FOR HIDAM/PHIDAM FOR HDAM/PHDAM FOR HISAM/SHISAM (MUST BE 008)

Explanation: An incorrect return code return_code was returned by the user exit routine. reason shows the reason why the return code is incorrect.

FOR HIDAM/PHIDAM:
Return code 16 is not allowed.

FOR HDAM/PHDAM:
Return code 20 is not allowed.

FOR HISAM/SHISAM:
Return codes 16 and 20 are not allowed.

(MUST BE 008):
Return code is not 8. The user exit routine must return code 8 for all the dependent segments of the segment that have return code 8.

System action: IMS HP Load issues a user 3707 abend and a dump.

User response: Correct the error and rerun the job.

INVALID SEGMENT LENGTH RETURNED FROM USER EXIT ROUTINE

Explanation: A user exit routine changed the length of a database segment. The modified length is larger than the maximum segment length allowed.

System action: IMS HP Load issues a user 3708 abend and a dump.

User response: Correct the error and rerun the job.

INVALID SEGMENT LENGTH RETURNED FROM COMPRESSION ROUTINE SEGMENT NAME: segment_name COMPRESSED LENGTH: nnnnn ALLOWED MAX LENGTH: nnnnn

Explanation: The length of a database segment that is returned from the segment edit/compression exit routine is larger than the allowed maximum value.

System action: A user 3709 abend and a dump are issued.

User response: Increase the maximum number of bytes for the reported segment, or correct the DBD or the segment edit/compression exit routine. Then, rerun the job.

DUMMY SPECIFIED FOR OVERFLOW DD

Explanation: The DUMMY parameter is specified in OVERFLOW DD. The DUMMY parameter is not allowed for OVERFLOW DD.

System action: IMS HP Load issues a user 3710 abend and a dump.

User response: Ensure that a DD statement is present for the ddname indicated, and that it properly identifies the correct data set. Correct the error and rerun the job.

DEVTYPE MACRO FAILURE OCCURRED FOR DDNAME: ddname RETURN CODE IS return_code

Explanation: After issuing a DEVTYPE macro to get information about the device associated with the ddname, the return code indicated that the attempt to
do so was unsuccessful. Return_code shows the return code.

**System action:** IMS HP Load issues a user 3711 abend and a dump.

**User response:** Ensure that the DD statement of indicated ddname indicates the correct data set. Correct the error and rerun the job.

---

**HPSR3712E DD NOT FOUND FOR DDNAME: ddname**

**Explanation:** The JCL contains no ddname DD statement. Such a statement is required.

**System action:** A user 3712 abend and a dump are issued.

**User response:** Ensure that the ddname DD statement exists in the JCL. Correct the error and rerun the job.

---

**HPSR3713E WORK AREA FOR USER EXIT user_exit IS TOO LARGE**

**Explanation:** The size of the work area for user exit is too large for the prepared work area by IMS HP Load.

**System action:** IMS HP Load issues a user 3712 abend and a dump.

**User response:** Modify the size of the work area that is defined in the user exit to reflect the maximum segment length. The size must be larger than the maximum segment length and smaller than the maximum segment length plus 1024 bytes.

---

**HPSR3714E DD NOT FOUND FOR SEQERROR**

**Explanation:** The JCL contains no DD statement for SEQERROR. If SEQERROR=SAVE is specified in the FRRIN or HPSIN DD statement, a SEQERROR DD statement is required.

**System action:** IMS HP Load issues a user 3714 abend.

**User response:** Ensure that the SEQERROR DD statement exists in the JCL. Correct the error and rerun the job.

---

**HPSR3715E SECONDARY INDEX MAINTENANCE EXIT ROUTINE routine IS NOT FOUND**

**Explanation:** The secondary index maintenance exit routine routine is not found.

**System action:** IMS HP Load issues a user 3715 abend and a dump.

**User response:** Ensure that the correct secondary index maintenance exit program library is concatenated to the STEPLIB DD.

---

**HPSR3716E SECONDARY INDEX MAINTENANCE EXIT ROUTINE routine LOAD FAILURE OCCURRED**

**Explanation:** IMS HP Load failed to load the secondary index maintenance exit routine routine.

**System action:** IMS HP Load issues a user 3716 abend and a dump.

**User response:** Contact IBM Software Support.

---

**HPSR3717E INVALID RETURN CODE FROM routine, RETURN CODE=return_code RETURN CODE GREATER THAN 999**

**Explanation:** An incorrect return code return_code or a value that is greater than 999 is returned by the indicated routine that is specified on the USERHDRX control statement. This message is also issued if the routine returns return code 4 and does not set the 'No Header/Trailer Record' indicator.

**System action:** IMS HP Load issues a user 3717 abend and a dump.

**User response:** Correct the error and rerun the job.

---

**HPSR3718E DFSURWF1 DD STATEMENT MISSING**

**Explanation:** Although the database that was reloaded is involved in a logical relationship or in a secondary index, DFSURWF1 DD is not specified.

**System action:** IMS HP Load issues a user 3718 abend and a dump.

**User response:** Check the DBD and the PSB to see if it is necessary to resolve a logical or secondary index relationship. Specify the DFSURWF1 data set, and rerun the job.

---

**HPSR3719E THE STORAGE AREA JUST PRIOR TO THE WORK AREA FOR USER EXIT HAS BEEN BROKEN**

**Explanation:** The storage area immediately before the work area for user exit is broken.

**System action:** IMS HP Load issues a user 3719 abend and a dump.

**User response:** Contact IBM Software Support.

---

**HPSR3720I DBDGEN REQUIRED FOR DATABASE "dbdname" TO SET DATXEXIT INDICATOR**

**Explanation:** IMS HP Load called the DFSDBUX1 exit routine and found that some of the database segments required conversion, but the DATXEXIT=YES parameter is not specified in the DBD dbdname. This
message is an information message to the user that a
DBDGEN with DATXEXIT=YES needs to be done for
this database.

System action: IMS HP Load continues processing.

User response: The database administrator needs to
be notified that a DBDGEN is required for this
database.

HPSR3721E SPECIFIED DATABASE "database" IS
NOT A TARGET FOR CONVERSION

Explanation: While processing database "database", for
which data conversion is specified by the
DATXEXIT=YES parameter, IMS HP Load called the
DFSDBUX1 exit routine and found that none of the
segments required conversion.

System action: IMS HP Load issues a user 3721 abend
and a dump.

User response: Correct the error and rerun the job.

HPSR3722E LOAD FAILED FOR DFSDBUX1
ROUTINE

Explanation: While processing a database that requires
data conversion by the DATXEXIT=YES parameter in a
DBD, IMS HP Load attempted to load the DFSDBUX1
exit routine but could not find the load module.

System action: IMS HP Load issues a user 3722 abend
and a dump.

User response: Correct the error and rerun the job.

HPSR3723E NONZERO RETURN CODE FROM
DFSDBUX1 ROUTINE RC = return_code

Explanation: A nonzero return code return_code is
returned by the DFSDBUX1 exit routine.

System action: IMS HP Load issues a user 3723 abend
and a dump.

User response: Correct the error and rerun the job.

HPSR3724E PARTITION SELECTION FAILURE
OCCURRED; REQUEST=xxxxxx
RC=return_code, RSN=reason_code

Explanation: For HALDB reload, the DL/I DFSPSEL
macro returned an error return code. xxxxxxx is the
request type of partition selection (the PART=
parameter value for the DFSPSEL macro), return_code is
the return code, and reason_code is the reason code. For
the explanation of these reason codes, see IMS Messages
and Codes.

System action: IMS HP Load issues a user 3724 abend
and a dump.

User response: If the corresponding IMS message
(DFS3nnnn) is issued, see the user response of the IMS
message or the following table. Then, correct the error
and rerun the job.

<table>
<thead>
<tr>
<th>Return code (return_code)</th>
<th>Reason code (reason_code)</th>
<th>User response</th>
</tr>
</thead>
<tbody>
<tr>
<td>04</td>
<td>4020</td>
<td>Database auth error occurs. If the corresponding IMS message (DFS3nnnn) is not issued, the possible cause of the error is a mismatch between the DSN registered with the RECON and the DSN that is processed by IMS HP Load. Check for any mismatch.</td>
</tr>
<tr>
<td>08</td>
<td>8011</td>
<td>An incorrect key value is passed to the DL/I DFSPSEL macro. Register 6 contains the passed key address. Check the key value and the high-key value of each partition.</td>
</tr>
<tr>
<td>8031</td>
<td></td>
<td>If the database data set is a VSAM data set with no REUSE parameter specified, this reason code is returned. The REUSE parameter is required for the HADLB data sets. Check the attribute of the database data set. Any other code</td>
</tr>
</tbody>
</table>

If necessary, contact IBM Software Support.

HPSR3725E ILE CREATION FAILURE OCCURRED;
REQUEST=xxxxxx, RC=return_code,
RSN=reason_code

Explanation: For HALDB reload, the DL/I DFSPCSH
macro returned an error return code. xxxxxxx is the
requested function (the FUNC= parameter value for the
DFSPCSH macro), return_code is the return code, and
reason_code is the reason code.

System action: IMS HP Load issues a user 3725 abend
and a dump.

User response: Correct the error and rerun the job. If
necessary, contact IBM Software Support.

HPSR3726E UNLOAD DATA SET REQUIRES
PREPROCESS FOR HALDB

Explanation: Segment records in the input unloaded
data set are out of the partition sequence. PSSR process
is required.
System action: IMS HP Load issues a user 3726 abend and a dump.

User response: Sort the records in the unloaded data set by using PSSR.

HPSR3727E LOAD FAILED FOR HPSRxxx7 ROUTINE

Explanation: Load failure occurred for the HPSRxxx7 module.

System action: IMS HP Load issues a user 3727 abend and a dump.

User response: Check the contents of the load module library of IMS HP Load. Correct the error and rerun the job.

HPSR3728E UNSUPPORTED RELEASE OF DFP

Explanation: The IMS HP Load job is run under an unsupported release of MVS/DFP. The DFSMSdfp of DFSMS/MVS Version 1 Release 1 or later is necessary to run the job.

System action: IMS HP Load issues a user 3728 abend and a dump.

User response: Run the job in the correct DFSMSdfp environment.

HPSR3729E UNSUPPORTED DBD:IMS/ESA PARTITION SUPPORT PRODUCT

Explanation: A DBD created with IMS Partition Support Product is not supported by IMS HP Load.

System action: IMS HP Load issues a user 3729 abend.

User response: Use the utilities that are provided by IMS Partition Support Product.

HPSR3730E LOAD FAILED FOR "module"

Explanation: Load failed for the indicated module.

System action: IMS HP Load issues a user 3730 abend and a dump.

User response: Check the contents of the load module library. Correct the error and rerun the job.

HPSR3731E CEEPIPI ERROR OCCURRED.

Function of CEEPIPI is init_sub or identify_entry. When function shows CALLSUB, the function of CEEPIPI is call_sub. For the return codes, see the topic about using preinitialization services in the Language Environment Programming Guide (SA22-7561).

Correct the error and rerun the job.

HPSR3732E SEGMENT LENGTH FIELD IS INCORRECT

Explanation: The length field of the segment contains a value greater than the allowed maximum or less than the allowed minimum.

System action: Module HPSRHDLD or HPSRRLDML issues a user 3732 abend and a dump.

User response: Correct the error and rerun the job.

HPSR3733E LOAD FAILED FOR MODULE: module

Explanation: An error was encountered while the module named module was being loaded.

System action: The Reload task issues a user 3733 abend.

User response: Ensure that the correct load module libraries are specified in the JOBLIB or STEPLIB concatenation. Also ensure that all required maintenance are applied. Correct the error and rerun the job. If necessary, contact IBM Software Support.

HPSR3734E NONZERO RETURN CODE FROM DFSRVSP0, RC=return_code

Explanation: An error occurred when DFSRVSP0 was called. Return code from DFSRVSP0 was return_code.

System action: The Reload task issues user 3734 abend.

User response: This error is likely an internal system error. Contact IBM Software Support.

HPSR3735E NONZERO RETURN CODE FROM DFSURIT0; error_type, PID=partition_id, RC=return_code, RSN=reason_code, CC=comp_code

Explanation: An error was found when DFSURIT0 was called to end the process for ILDS of Partition ID: partition_id. The DFSURIT0 either ended with a return code of return_code and a reason code of reason_code or ended abnormally with a completion code of comp_code.

<table>
<thead>
<tr>
<th>error_type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>GETMAIN error</td>
</tr>
<tr>
<td>02</td>
<td>FREEMAIN error</td>
</tr>
<tr>
<td>03</td>
<td>VSAM error</td>
</tr>
<tr>
<td>04</td>
<td>Load error</td>
</tr>
</tbody>
</table>

Chapter 19. Messages and codes 285
<table>
<thead>
<tr>
<th>error_type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>05</td>
<td>Sort error</td>
</tr>
<tr>
<td>06</td>
<td>Data space error</td>
</tr>
<tr>
<td>07</td>
<td>DFSURIT0 ended abnormally</td>
</tr>
</tbody>
</table>

**System action:** The Reload task issues a user 3735 abend.

**User response:** For the data space error (error_type=06), the return code and the reason code that are shown in the message are from the DSPSERV macro. See the MVS Assembler Services Reference and check the meaning. Otherwise, this error is likely to be an internal system error. Record the message and contact IBM Software Support.

**Explanation:** DBRC is called to create a REORG record for the database, but the data set name of ddname is not registered with DBRC.

**System action:** IMS HP Load issues a user 3736 abend and a dump.

**User response:** Ensure that the correct RECON data sets are used. Correct the error, and rerun the job.

**Explanation:** The allocation information of the data set to be reloaded does not meet the specification of IMS.

xxxxxx is one of the following values:

- CI
- BLOCK
- RECORD

<table>
<thead>
<tr>
<th>Reason</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CI SIZE &gt; 30720</td>
<td>The specified CI size exceeds 30720.</td>
</tr>
<tr>
<td>CI SIZE NOT EQUAL LRECL+7</td>
<td>The CI size must be equal to the RECORD size plus 7 bytes.</td>
</tr>
<tr>
<td>BLOCK SIZE &gt; 32767</td>
<td>The specified BLOCK size exceeds 32767.</td>
</tr>
<tr>
<td>RECORD SIZE &gt; 32760</td>
<td>The specified RECORD size exceeds 32760.</td>
</tr>
<tr>
<td>RECORD SIZE IS ODD-SIZE</td>
<td>The RECORD size cannot be an odd value.</td>
</tr>
</tbody>
</table>

**System action:** IMS HP Load issues a user 3737 abend.

**User response:** Correct the error, and rerun the job.

**Explanation:** Catalog Search Interface IGGCS100 caused an error. text describes the error.

**text represents:**

- LINK ERROR
- RETURN CODE ERROR
- ERROR OCCUR FOR ENTRY

**System action:** Module HPSRCL1 issues a user 3738 abend and a dump.

**User response:** Contact IBM Software Support.

To improve the performance of rebuilding ILDS by making the VSAM load mode in effect, specify the REUSE parameter in your ILDS CLUSTER definition, and specify the ILDSBLD=YES control statement.

**Explanation:** The extension number of a reorganized data set is larger than the number that you specified for EXTENT=.

**System action:** A user 3739 abend and a dump are issued.

**User response:** Check the SPACE parameter in the allocation information of the OSAM data set, and increase the allocation size of the primary or secondary definition if needed. Rerun the job.

**Explanation:** An error occurred when HPSRFIG1 was called. Return code from HPSRFIG1 was return_code.

**System action:** The Reload task issues a user 3740 abend.

**User response:** This problem is likely to be an internal system error. Contact IBM Software Support.

**Explanation:** Data set name for DD name xxxxxxx is not registered to DBRC.

**System action:** IMS HP Load issues a user 3741 abend.

**User response:** Correct the error and rerun the job.
HPSR3742W AN INTERNAL xxxxxxxxxx COMMAND FAILURE OCCURRED.

Explanation: An error was encountered while the xxxxxxxxxx command was running for DSG check.

System action: Processing continues; DSG check was ignored.

User response: This problem is an internal system error. If necessary, contact IBM Software Support.

HPSR3743W DYNALLOC FAILURE OCCURRED FOR xxxxxxxx RETURN CODE=return_code REASON CODE=reason_code

Explanation: An attempt to dynamically allocate data set xxxxxxxx failed. xxxxxxxx is used for the DSG check or BLKSIZE check internally. Return_code and reason_code are the return code and the reason code from the SVC99 routine.

System action: Processing continues; DSG check or BLKSIZE check is bypassed. When DSG check is bypassed, message HPSR3744W follows. When BLKSIZE check is bypassed, message HPSR3699W follows.

User response: If necessary, correct the error and rerun the job or contact IBM Software Support.

HPSR3744W xxxxxxx OPEN FAILURE OCCURRED

Explanation: An attempt to open the data set xxxxxxxx failed. xxxxxxxx is used for DSG check internally.

System action: Processing continues; DSG check was ignored.

User response: If necessary, correct the error and rerun the job or contact IBM Software Support.

HPSR3745W LOAD FAILED FOR MODULE: module

Explanation: An error was encountered while the module named module was being loaded.

System action: Processing continues; DSG check was ignored.

User response: If necessary, correct the error and rerun the job or contact IBM Software Support.

HPSR3746I ALLOC RECORD FOUND FOR DBD: dbdname DDN: ddname TIMESTAMP: yyyygdddmhmsst

Explanation: An ALLOC record is found for the database dbdname DD name ddname with the time stamp yyyygdddmhmsst which is newer than that of the unloaded data set creation.

System action: Processing continues.

User response: None. This message is informational.

HPSR3747E ABEND REQUESTED BY UPDATE CHECK PROCESS

Explanation: An ALLOC record is found for the database with the time stamp that is newer than that of the unloaded data set creation. The job step ended abnormally as requested by the CHKUPDT control statement.

System action: IMS HP Load issues a user 3747 abend and a dump.

User response: Ensure that you are not using a back level unloaded data set mistakenly. Correct the error and rerun the job.

HPSR3748I UPDATE CHECK NOT PROCESSED. REASON CODE: reason_code

Explanation: A check for DBRC to verify that there are no updates between unload and reload processes cannot be done.

<table>
<thead>
<tr>
<th>Reason code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>DBRC is not active.</td>
</tr>
<tr>
<td>02</td>
<td>DBDS record is not registered to DBRC RECON.</td>
</tr>
<tr>
<td>03</td>
<td>Format of unloaded data set is not either of HD, SH, or PHD.</td>
</tr>
<tr>
<td>04</td>
<td>Timestamp is not found in the unloaded data set.</td>
</tr>
</tbody>
</table>

System action: Processing continues. No check is done for an update between unload and reload.

User response: If you want to check an update, correct the error and rerun the job.

HPSR3749E AN INTERNAL dbrcmd COMMAND FAILURE OCCURRED. RC: return_code

Explanation: An internal DBRC command dbrcmd failed with return code return_code.

System action: IMS HP Load issues a user 3749 abend and a dump.

User response: Correct the error and rerun the job.

HPSR3750E HPIO INIT CALL ERROR

Explanation: The INIT call for the High Performance I/O Driver failed.

System action: IMS HP Load issues a user 3750 abend and a dump.

User response: Register 15 contains the address of the diagnostic information.
If necessary, contact IBM Software Support.

HPSR3751E  HPIO OPEN CALL ERROR
Explanation: The OPEN call for the High Performance I/O Driver failed.
System action: IMS HP Load issues a user 3751 abend and a dump.
User response: Register 15 contains the address of the diagnostic information.
If necessary, contact IBM Software Support.

HPSR3752E  CLUSTER NOT EMPTY
Explanation: The OPEN call for the High Performance I/O Driver found that the specified database data set was not empty.
System action: IMS HP Load issues a user 3752 abend and a dump.
User response: Correct the error and rerun the job.

HPSR3753E  HPIO POINT CALL ERROR
Explanation: The POINT call for the High Performance I/O Driver failed.
System action: IMS HP Load issues a user 3753 abend and a dump.
User response: Register 15 contains the address of the diagnostic information.
If necessary, contact IBM Software Support.

HPSR3754E  HPIO WRITE CALL ERROR
Explanation: The WRITE call for the High Performance I/O Driver failed.
System action: IMS HP Load issues a user 3754 abend and a dump.
User response: Register 15 contains the address of the diagnostic information.
If necessary, contact IBM Software Support.

HPSR3755E  HPIO CLOSE CALL ERROR
Explanation: The CLOSE call for the High Performance I/O Driver failed.
System action: IMS HP Load issues a user 3755 abend and a dump.
User response: Register 15 contains the address of the diagnostic information.
If necessary, contact IBM Software Support.

HPSR3756E  HPIO EXTEND FAILURE
Explanation: The High Performance I/O Driver detected an extend error for the database data set. There is a shortage of space.
System action: IMS HP Load issues a user 3756 abend and a dump.
User response: Correct the error and rerun the job.

HPSR3757E  HPIO TERM CALL ERROR
Explanation: The TERM call for the High Performance I/O Driver failed.
System action: IMS HP Load issues a user 3757 abend and a dump.
User response: Register 15 contains the address of the diagnostic information.
If necessary, contact IBM Software Support.

HPSR3758E  HPIO GETBUF CALL ERROR
Explanation: The GETBUF call for the High Performance I/O Driver failed.
System action: IMS HP Load issues a user 3758 abend and a dump.
User response: Register 15 contains the address of the diagnostic information.
If necessary, contact IBM Software Support.

HPSR3759I  HPIO IS NOT USED BECAUSE AT LEAST ONE LIBRARY IS NOT AUTHORIZED
Explanation: High Performance I/O (HPIO) is not used in subtasks because at least one library is not APF authorized.
System action: IMS HP Load continues processing.
User response: If you want to use HPIO, you must APF-authorize all libraries that are in the STEPLIB DD concatenation.

HPSR3760E  LOAD FAILED FOR "module"
Explanation: Load failed for the indicated module.
System action: IMS HP Load issues a user 3760 abend and a dump.
User response: Check the contents of the load module library. Correct the error and rerun the job.
HPSR3761E  LIBRARY NOT AUTHORIZED
Explanation: A load library of STEPLIB is not APF-authorized.
System action: IMS HP Load sets a return code of 8.
User response: Correct the error and rerun the job.

HPSR3762E  OSAM DATA SET OPEN FAILURE OCCURRED
Explanation: An error occurred while an OSAM data set was being opened.
System action: Module HPSROSAM issues a user 3762 abend and a dump.
User response: If you specify DBALLABOVE in DFSVSAMP and message IEC133I also appeared in the job log, make the following change and rerun the job:
• For z/OS 1.12 or later, specify NON_VSAM_XTIOT=YES in the DEVSUP_xx PARMLIB member.
• For z/OS 1.11 or earlier, remove the specification of DBALLABOVE from DFSVSAMP.

HPSR3763E  PARTITION ASSIGNED IS OUTSIDE THE RANGE FOR REORGANIZATION
Explanation: The assigned partition is outside the range of the partitions that are specified by the PARTITION control statement in the REORG block.
System action: Module HPSRRLDLM issues a user 3763 abend.
User response: Review the PARTITION control statement and rerun the job.

HPSR3764E  KEY FIELD IS NOT DEFINED FOR THE ROOT SEGMENT OF HALDB
Explanation: IMS HP Load found no sequence field in the root segment of the HALDB.
System action: IMS HP Load issues a user 3764 abend.
User response: Correct the error and rerun the job.

HPSR3765E  IDCAMS COULD NOT DEFINE A GDG
Explanation: IDCAMS failed in defining a GDG. The error messages that are issued by IDCAMS are shown in the GDG Base Information report.
System action: IMS HP Load issues a user 3765 abend.
User response: Locate the IDCAMS messages, correct any errors, and rerun the job.

HPSR3766E  GDG BASE ENTRY FOR DFSURWF1 WAS NOT FOUND (R15=register15, R0=register0)
Explanation: IMS HP Load tried to create a generation data set. However, the GDG base entry was not found by the LOCATE macro. register15 and register0 show the returned information from the LOCATE macro.
System action: IMS HP Load issues a user 3766 abend.
User response: Ensure that the specifications of WF1DSN and GDGBASE are correct. If you specify a symbolic parameter &GDG on the WF1DSN statement and if you want to create a DFSURWF1 data set together with a new generation data set group, specify GDGBASE=YES. Otherwise, contact IBM Software Support.

HPSR3767E  THE IDCAMS PROGRAM RETURNED AN ERROR
Explanation: The IDCAMS program that is internally called by IMS HP Load returned an error return code. The detailed error messages are shown in the Database Data Set Allocation report.
System action: IMS HP Load issues a user 3767 abend.
User response: Locate the IDCAMS messages in the report, correct any errors, and rerun the job.

HPSR3768W  THE RELOADED PARTITION (partition) IS NOT SPECIFIED ON THE LPART STATEMENT
Explanation: The partitions to be reloaded are specified on the LPART control statement. However, IMS HP Load reloaded the indicated partition, which is not specified on the LPART control statement.
System action: IMS HP Load continues processing.
User response: Ensure that the LPART statement contains all the partitions that you want to reload. If necessary, correct the LPART statement and rerun the job.

HPSR3769E  PARTINIT=YES,LPART CANNOT BE SPECIFIED BECAUSE LPART STATEMENT IS NOT SPECIFIED
Explanation: PARTINIT=YES,LPART is specified; however, the LPART control statement is not specified. To specify PARTINIT=YES,LPART, you must also specify the LPART control statement.
System action: IMS HP Load issues a user 3769 abend.
User response: Correct the control statement and rerun the job.
HPSR3770E • HPSR3780E

HPSR3770E  SORT PROCESSING ENDED ABNORMALLY

Explanation: Sort processing ended abnormally.
System action: Module HPSRSRT0 issues a user 3770 abend.
User response: See the accompanying error messages.

HPSR3771E  DO NOT USE A LARGE FORMAT D/S: dname

Explanation: The dname data set is a large format data set that is supported by z/OS Version 1 Release 7 and later. IMS HP Load cannot process large format data sets in z/OS Version 1 Release 6 or that of prior versions.
System action: IMS HP Load issues a user 3771 abend and a dump.
User response: Check the level of z/OS.

HPSR3772E  GETDSAB FAILED FOR dname

Explanation: The GETDSAB macro failed for DD name dname.
System action: IMS HP Load issues a user 3772 abend and a dump.
User response: This problem is likely to be an internal system error. Contact IBM Software Support.

HPSR3773E  SWAREQ FAILED FOR dname

Explanation: The SWAREQ macro failed for DD name dname.
System action: IMS HP Load issues a user 3773 abend and a dump.
User response: This problem is likely to be an internal system error. Contact IBM Software Support.

HPSR3774E  OBTAIN FAILED FOR dname

Explanation: The OBTAIN macro failed for DD name dname.
System action: IMS HP Load issues a user 3774 abend and a dump.
User response: This problem is likely to be an internal system error. Contact IBM Software Support.

HPSR3775E  DBRC COMMAND PROCESSING ENDED ABNORMALLY

Explanation: DBRC command processing ended abnormally. An error occurred while detaching a DBRC command task.
System action: IMS HP Load issues a user 3775 abend.
User response: Contact IBM Software Support.

HPSR3777E  DDEFPDS CANNOT BE USED BECAUSE AT LEAST ONE LIBRARY IS NOT AUTHORIZED

Explanation: Database data sets cannot be created by using the DDEFPDS DD statement because at least one library is not APF authorized.
System action: IMS HP Load issues a user 3777 abend.
User response: To use the IDCAMS commands with the DDEFPDS DD statement, APF-authorize all the libraries that are in the STEPLIB DD concatenation, and rerun the job.

HPSR3778E  SEGMENT LENGTH IN PSINDEX DB (DBDNAME=ddbname) IS TOO LONG

Explanation: IMS High Performance Load cannot create WFP data sets because the segment length that is defined in the DBD definition of the indicated database is too long.
System action: IMS HP Load issues a user 3779 abend and a dump.
User response: Remove the WFPHLQ control statement, or make the sum of the segment length and the root key length of the PSINDEX less or equal to 7189.

HPSR3779E  WFP DATA SET (DDNAME=ddbname) FOR partition IS NOT EMPTY

Explanation: The data set that is specified by the indicated DD name is not empty. The indicated DD name is one of the DD names that are used to specify the WFP data sets for the indicated partition.
System action: IMS HP Load issues a user 3779 abend and a dump.
User response: Ensure that the correct data set is specified on the indicated DD name.

HPSR3780E  DUMMY SPECIFIED FOR dname DD

Explanation: The DUMMY parameter is specified for the indicated DD name. However, the DUMMY parameter is not allowed for this DD statement.
System action: IMS HP Load issues a user 3780 abend and a dump.
User response: Ensure that the DD statement exists and that it specifies the correct data set. Correct the error and rerun the job.
HPSR3781E  WFP DATA SET NAME (DDNAME=ddname) FOR partition IS INCORRECT

Explanation: The data set name that is assigned by the indicated DD name is incorrect. WFP data set names must follow the naming convention.

System action: IMS HP Load issues a user 3781 abend and a dump.

User response: Ensure that the correct data set name is specified on the DD statement.

HPSR3782E  LOCATE FAILED(R15=register15,R0=register0), DSN=ddname

Explanation: An error occurred while the LOCATE macro was processing the indicated data set. register15 and register0 show the returned information from the LOCATE macro.

System action: IMS HP Load issues a user 3782 abend and a dump.

User response: See z/OS DFSMS Managing Catalogs to interpret the returned information from the LOCATE macro. Correct the error and rerun the job. If the error persists, contact IBM Software Support.

HPSR3783E  DBDS CANNOT BE EXTENDED DUE TO THE LIMIT OF RBA

Explanation: IMS HP Load attempted to extend the VSAM database data set, but the data set could not be extended because the high-allocated RBA reached the 4 GB size limit.

System action: Module HPSRIOIN issues a user 0004 abend.

User response: Decrease the total size of the database by one of the following methods:

- Archive old data.
- Use segment compression.
- Divide the database data set into two or more data set groups.
- Convert to OSAM or HALDB.

If you are reloading an HDAM or a PHDAM database, you can also decrease the total size of the database by one of the following methods:

- Adjust the SEARCH control statement value to decrease the number of segments that will be directed to the overflow portion of the database.
- Decrease the number of RAA blocks if there are many unused RAPs and the RAA has a high percentage of free space.

If necessary, contact IBM Software Support.

HPSR3784E  VSAM EXTEND FAILURE OCCURRED IN HPIO

Explanation: An error occurred while extending the VSAM data set (EXTEND VSAM) with HPIO. This error could be the result from one of the following causes:

- The new allocation amount exceeds the maximum RBA.
- All space is used on the current volume and no more volumes are specified.

For more information, see the IEC message that is printed before this message.

System action: Module HPSRIOIN issues a user 0004 abend.

User response: If the new allocation amount exceeds the maximum RBA, decrease the total size of the database by one of the following methods:

- Archive old data.
- Use segment compression.
- Divide the database data set into two or more data set groups.
- Convert to OSAM or HALDB.

If you are reloading an HDAM or a PHDAM database, you can also decrease the total size of the database by one of the following methods:

- Adjust the SEARCH control statement value to decrease the number of segments that will be directed to the overflow portion of the database.
- Decrease the number of RAA blocks if there are many unused RAPs and the RAA has a high percentage of free space.

HPSR3801E  PARAMETER ERROR DETECTED ON PARM: parm_string

Explanation: An incorrect parameter is detected. The incorrect parameter string is listed as parm_string.

System action: The IPR Reload utility issues a user 3801 abend.

User response: Check the parameter specification that is listed as parm_string. Correct the parameter, and rerun the job.

HPSR3810E  OPEN FAILED FOR DDNAME: ddname

Explanation: After an OPEN macro was issued to open the non-VSAM data set that is associated with ddname, DCBOFLGS was not zero.

System action: IMS HP Load issues a user 3810 abend.

User response: Ensure that the DD statement
HPSR3811E  •  HPSR3818E

associated with ddname specifies the correct data set. Correct the error and rerun the job.

HPSR3811E  •  BLDL MACRO FAILED FOR MEMBER
member IN LIBRARY ddname
(RC=return_code, RSN=reason_code)

Explanation: The BLDL macro for member member in library ddname failed. The return code form the macro call is return_code, and the reason code is reason_code.

System action: The IPR Reload utility issues user 3811 abend.

User response: This error is likely an internal system error. Contact IBM Software Support.

HPSR3812E  •  OPEN FAILED FOR DDNAME: ddname
(ACBERFLG=xxx)

Explanation: When an OPEN macro was issued to open the VSAM data set that is associated with ddname, an error occurred. The value xxx is the decimal OPEN reason code in the ACBERFLG field of the ACB for the data set in error.

System action: The Reload task issues user 3812 abend.

User response: Check the reason code in the ACBERFLG field. The reason codes are explained in DFSMS/MVS Macro Instructions for Data Sets. Correct the error and rerun the job.

If necessary, contact IBM Software Support.

HPSR3813E  •  CLOSE FAILED FOR DDNAME: ddname
(ACBERFLG=xxx)

Explanation: An error occurred while the VSAM data set that is allocated to the DD ddname was being closed. The value xxx is the decimal CLOSE reason code in the ACBERFLG field of the ACB for the data set in error.

System action: The Reload task issues user 3813 abend.

User response: Check the reason code in the ACBERFLG field. The reason codes are explained in DFSMS/MVS Macro Instructions for Data Sets. Correct the error and rerun the job.

If necessary, contact IBM Software Support.

HPSR3814E  •  VSAM GET FAILED FOR DDNAME:
ddname (RC=return_code,
RSN=reason_code)

Explanation: An error occurred when a VSAM GET was issued for the VSAM data set that is allocated to DD ddname. The value return_code is the return code in the RPLRTNCD field of the RPL used for the VSAM GET request. The value reason_code is the decimal reason code in the RPLERRCD field of the RPL.

System action: The Reload task issues a user 3814 abend.

User response: Check the return and reason codes. These codes are explained in DFSMS/MVS Macro Instructions for Data Sets. Correct the error and rerun the job.

If necessary, contact IBM Software Support.

HPSR3815E  •  LOAD FAILED FOR MODULE:
mod_name DDNAME: ddname

Explanation: After a LOAD macro was issued to load the module mod_name from the library that is specified by the ddname DD statement, register 15 contained a nonzero return code.

System action: IMS HP Load issues a user 3815 abend.

User response: Ensure that the DD statement associated with ddname specifies the correct data set. Correct the error and rerun the job. If the problem persists, save the entire run listing (including the dump and the JCL) and contact IBM Software Support.

HPSR3816E  •  UNSUCCESSFUL VSAM SHOWCB
FOR ACB

Explanation: After a SHOWCB macro was issued for an ACB control block, register 15 contained a nonzero return code.

System action: The Reload task issues a user 3816 abend.

User response: This error is likely an internal system error. Contact IBM Software Support.

HPSR3817E  •  UNSUCCESSFUL VSAM SHOWCB
FOR RPL

Explanation: After a SHOWCB macro was issued for an RPL control block, register 15 contained a nonzero return code.

System action: The Reload task issues a user 3817 abend.

User response: This error is likely an internal system error. Contact IBM Software Support.

HPSR3818E  •  ERROR RETURNED FROM IGWASYS
(RC=return_code, RSN=reason_code)

Explanation: IMS HP Load called IGWASYS, but received an error return code from IGWASYS. Here, RC shows the return code, and RSN shows the reason code.

System action: IMS HP Load issues a user 3818 abend and a dump.

User response: See DFSMSdfp Advanced Services and
check the meaning of the return code and reason code of IGWASYS. Correct the error and rerun the job.

**HPSR3819E**  RDJFCB MACRO FAILED ON DDNAME: ddbname (RC=return_code)  

**Explanation:** The RDJFCB macro failed for DD name ddbname. The return code from the macro call is return_code.  

**System action:** IMS HP Load issues a user 3819 abend and a dump.  

**User response:** This problem is likely an internal system error. Contact IBM Software Support.

**HPSR3820E**  DYNAMIC ALLOCATION FAILURE OCCURRED FOR DB: dbdname DD: ddbname. RC=return_code RSN=reason_code  

**Explanation:** An attempt at dynamic allocation for ddbname of dbdname failed. Return_code is the return code in register 15, and reason_code is the associated hexadecimal reason code from the SVC99 routine.  

**System action:** IMS HP Load issues a user 3820 abend.  

**User response:** The return code and reason code from the SVC99 routine are explained in the MVS Authorized Assembler Services Guide. Correct the error and rerun the job.

**HPSR3821E**  VSAM ERASE FAILED FOR DDNAME: ddbname (RC=return_code, RSN=reason_code)  

**Explanation:** An error occurred when a VSAM ERASE command was issued for the VSAM data set that is allocated to DD ddbname. Value return_code is the return code in the RPLRTNCD field of the RPL used for the VSAM ERASE request. Value reason_code is the decimal reason code in the RPLERRCD field of the RPL.  

**System action:** The Reload task issues a user 3821 abend.  

**User response:** Check the return and reason codes. These codes are explained in DFSMS/MVS Macro Instructions for Data Sets. Correct the error and rerun the job.  

If necessary, contact IBM Software Support.

**HPSR3822E**  CSVQUERY MACRO FAILED FOR entry_name (RC=return_code)  

**Explanation:** The CSVQUERY macro failed for a name of entry point: entry_name. RC shows the return code from the macro call.  

**System action:** IMS HP Load issues a user 3822 abend and a dump.  

**User response:** This problem is likely an internal system error. Contact IBM Software Support.

**HPSR3823E**  ATTACHX MACRO FAILED FOR module (RC=return_code)  

**Explanation:** The ATTACHX macro for module failed. RC shows the return code from the macro call.  

**System action:** IMS HP Load issues a user 3823 abend and a dump.  

**User response:** See MVS Auth Asm Services Reference ALE-DYN: 1.0 and check the meaning of the return code and the reason code of ATTACHX. If necessary, contact IBM Software Support.

**HPSR3825E**  DEVTYPE MACRO FAILED ON DDNAME: ddbname RC=return_code  

**Explanation:** After a DEVTYPE macro was issued to check the data set that is specified in the ddbname DD statement, the return code indicated that the attempt to do so was unsuccessful. Return_code is the DEVTYPE return code.  

**System action:** IMS HP Load issues a user 3825 abend.  

**User response:** Ensure that the DD statement associated with ddbname specifies the correct data set. Correct the error and rerun the job.

**HPSR3830E**  INCORRECT DFSMDA MEMBER: member  

**Explanation:** Member is loaded as a DFSMDA member, but it does not have a correct DFSMDA format. The eye-catching MDA is not found in the loaded member (member).  

**System action:** IMS HP Load issues a user 3830 abend.  

**User response:** If a library that contains a member with the same name member is specified in the higher order of the STEPLIB or JOBLIB concatenation, this problem can occur. Ensure that the correct DFSMDA library is concatenated to the STEPLIB or JOBLIB. Correct the error and rerun the job.

**HPSR3840E**  DD: dbdname INFORMATION OF DB: dbdname IS NOT FOUND IN DFSMDA MEMBER  

**Explanation:** The DD information that is associated with dbdname DD is not found in the DFSMDA member. The IPR Reload utility cannot attempt the dynamic allocation for the dbdname of database (dbdname).  

**System action:** IMS HP Load issues a user 3840 abend.
HPSR3841E  INVALID ALLOCATE REQUEST FOR DATABASE. DDN=dbname

Explanation: Some, but not all, of the data sets of a database were allocated by JCL. The entire database must be allocated by JCL or by the DFSMDA member, but not both. Mixed allocations are invalid.

System action: IMS HP Load issues a user 3841 abend.

User response: Correct the JCL and rerun the job.

HPSR3845E  DDNAME SPECIFIED IN DBD DOES NOT HAVE A CORRESPONDING MATCH WITH A DD STATEMENT

Explanation: The IPR Reload utility tried to allocate a database data set, but an MDA member that corresponds to the specified DBD or a DD statement for the DD name that is specified in the DBD was not found.

System action: IMS HP Load issues a user 3845 abend.

User response: Either add a DD statement for the database data set or specify (on the STEPLIB or JOBLIB) a PDS data set that contains an MDA member for the DBD.

HPSR3850E  LOAD-MODULE xxxxxxx IS NOT REENTRANT

Explanation: The IPR Reload utility requires that xxxxxxx be link-edited with the link-editor attribute RENT. This was not done.

System action: The IPR Reload utility issues a user 3850 abend.

User response: Correct the error and rerun the job. If the error persists, contact IBM Software Support.

HPSR3855E  NONZERO RETURN CODE FROM IDENTIFY: RC=return_code

Explanation: The IPR Reload utility issues an IDENTIFY macro. The IDENTIFY failed. This must not occur. Return_code is the IDENTIFY return code.

System action: The IPR Reload utility issues a user 3855 abend.

User response: Contact IBM Software Support.

HPSR3860E  HPSRFCA0 IS NOT REUSABLE

Explanation: The load module HPSRFCA0 does not have the link-editor attribute REUS, which is required.

System action: The runtime initializer ends abnormally.

User response: Identify why HPSRFCA0 or its submodule is not reusable. If necessary, contact IBM Software Support.

HPSR3865E  COMPRESSION EXIT ROUTINE

nnnnnn function ERROR - Uaaaa
REASON rrrrrrr

Explanation: An error was detected by a Segment Edit/Compression exit routine. In the message text:

function The function of the exit routine that caused the error.

Uaaaa The user abend code generated by the exit routine.

rrrrrr The unique label at which the error was detected. This label corresponds to the error reason code. Find the reason code in the user’s guide of the Segment Edit/Compression exit routine or contact the supplier of the Segment Edit/Compression exit routine.

System action: IMS HP Load issues a user 3865 abend and a dump.

User response: Determine the cause of the error and correct the problem.

HPSR3870E  NAME/TOKEN SERVICE xxxxxxx FAILED. NAME: nametoken RC=return_code

Explanation: The process failed in the z/OS MVS Name/Token Service. xxxxxxx shows the service name. Return_code is the return code of the Name/Token Service.

System action: IMS HP Load issues an user 3870 abend.

User response: Find the cause of the error. For the return code, see z/OS MVS Programming: Assembler Services Reference, Volume 2 (IARR2V-XCTLX). If necessary, correct the warning condition and rerun the job.

HPSR3871I  RDE CREATED FOR DBD/PART: dbdname/partname

Explanation: The registered DMB entry (RDE) is successfully created for database dbdname/partname.

System action: Processing continues.
User response: None. This message is informational.

HPSR3872W LIU API FUNCTION ERROR. 
DBD/PART=dbdname/partname 
FUNC=xxxxx RC=return_code 
RSN=reason_code

Explanation: The API function in the IMS Library Integrity Utilities (LIU), Integrity Checker ended with an error. The registered DMB entry (RDE) was not created or updated.

System action: Processing continues.

User response: Create the RDE after the reload process, or correct the warning condition and rerun the job.

HPSR3873E THE INPUT DATA IS COMPRESSED BY THE OLD COMP ROUTINE.

Explanation: The input data was compressed by a compression routine other than that is defined in the DBD. The data cannot be decompressed.

System action: IMS HP Load issues a user 3873 abend.

User response: If you change the compression routine, use a decompressed unload data set for input.

HPSR3874E THE INFORMATION ABOUT RDE CREATION WAS NOT FOUND. 
DBD/PART=dbdname/partname

Explanation: IMS HP Load must obtain the information about the creation of the registered DMB entry (RDE) from IMS Library Integrity Utilities (LIU). However, IMS HP Load could not find the information for database dbdname/partname.

System action: IMS HP Load issues a user 3874 abend.

User response: Contact IBM Software Support.

HPSR3875E DBRC REQUIRED FOR THIS EXECUTION

Explanation: The target database is a HALDB database or an IMS catalog database, but DBRC=N is specified in the JCL.

System action: IMS HP Load issues a user 3875 abend and a dump.

User response: Specify DBRC=Y and rerun the job.

If you want to process an IMS catalog database without using DBRC, complete either of the following steps and rerun the job:

- Ensure that the catalog definition exit routine (DFS3CDX0) is specified correctly. For more information, see the topic "IMS Catalog Definition exit routine (DFS3CDX0)" in IMS Exit Routines.

HPSR3876I COMPAUTH=YES CANNOT BE USED BECAUSE ONE OR MORE LIBRARIES ARE NOT AUTHORIZED

Explanation: IMS HP Load ignored COMPAUTH=YES because one or more libraries are not APF authorized.

System action: Processing continues; the keyword parameter is ignored.

User response: To use COMPAUTH=YES, APF-authorize all the libraries that are in the STEPLIB DD concatenation.

HPSR3901E OPEN FAILED FOR DDNAME: ddname

Explanation: After an OPEN macro was issued to open the non-VSAM data set that is associated with ddname, DCBOFLGS was not zero.

System action: Processing stops.

User response: Ensure that the DD statement associated with ddname indicates the correct data set. Correct the error and rerun the job.

HPSR3902E CONTROL STATEMENT SYNTAX ERROR

Explanation: An incorrect control statement is detected by the control statement syntax checking.

System action: Processing stops or IMS HP Load issues a user 3902 abend.

User response: Check the control statement. Correct the error and rerun the job.

HPSR3903E INVALID PARAMETER SPECIFIED: parm_string.

Explanation: An incorrect parameter is detected. The incorrect parameter string is listed as parm_string.

System action: Processing stops.

User response: Check the parameter specification that is listed as parm_string. Correct the parameter error, and rerun the job.

HPSR3905E NUMBER OF OPERANDS FOR parm_name PARAMETER EXCEEDED THE LIMIT: nnn

Explanation: An incorrect control statement is detected by the control statement syntax checking.

System action: Processing stops.
**HPSR3906E • HPSR4002W**

**User response:** Check the control statement. Correct the error and rerun the job.

---

**HPSR3906E LENGTH ERROR IN n-th OPERAND OF PARAMETER: parm_name**

**Explanation:** An incorrect control statement is detected by the control statement syntax checking.

**System action:** Processing stops.

**User response:** Check the control statement. Correct the error and rerun the job.

---

**HPSR3907E ATTRIBUTE ERROR IN n-th OPERAND OF PARAMETER: parm_name**

**Explanation:** An incorrect control statement is detected by the control statement syntax checking.

**System action:** Processing stops.

**User response:** Check the control statement. Correct the error and rerun the job.

---

**HPSR3908E NUMBER OF VALUES FOR n-th OPERAND EXCEEDED THE LIMIT: mnn**

**Explanation:** An incorrect control statement is detected by the control statement syntax checking.

**System action:** Processing stops.

**User response:** Check the control statement. Correct the error and rerun the job.

---

**HPSR3909E n-th OPERAND IS REQUIRED FOR PARAMETER: parm_name**

**Explanation:** An incorrect control statement is detected by the control statement syntax checking.

**System action:** Processing stops.

**User response:** Check the control statement. Correct the error and rerun the job.

---

**HPSR3910I BLANK LINE IN DDNAME: ddname**

**Explanation:** A blank line is read in the FRRIN or HPSIN data set.

**System action:** Processing continues.

**User response:** None. This message is informational.

---

**HPSR3911E "keyword" IS ONLY APPLIED TO IPR RELOAD.**

**Explanation:** Incorrect keyword is specified. The keyword is available only for the IPR Reload utility.

**System action:** Processing stops.

---

**HPSR3999E UNKNOWN ERROR OCCURRED IN module-name MODULE (CC = mnn)**

**Explanation:** An internal error occurred in the indicated module of IMS HP Load module-name.

**System action:** IMS HP Load issues a user 3999 abend.

**User response:** Save the entire listing (including the dump, JCL, and reports) and contact IBM Software Support.

---

**HPSR4000E LOAD API PROCESS FAILED.**

**Explanation:** The Load API processing failed when the application program processed the DL/I call to the PCB that is specified with the control statement. dbdname shows the name of the DBD to which the PCB refers to, mnnn shows the PCB number in the PSB, and pcbname shows the label of the PCB. If the PCB name is not specified in the PSB, it will be blank. For details, see the HPSRmnnnE and DFSmnnn messages that are issued before this message.

**System action:** IMS HP Load issues a user 4000 abend.

**User response:** See the accompanying error messages and follow the required user response sections for those messages. Correct the error and rerun the job.

---

**HPSR4001W TOO FEW PARAMETERS WERE DETECTED IN THE CALL(function)**

**Explanation:** Function shows the DL/I call function. The number of the specified parameters for this function is insufficient. The call is passed to IMS and the program abends.

**System action:** The task issues IMS abends.

**User response:** If necessary, check the reason why your application issues the ROLL call.

---

**HPSR4002W ROLL CALL IS ISSUED FROM APPLICATION PROGRAM.**

**Explanation:** IMS HP Load received a ROLL call from your application program.

**System action:** IMS HP Load passes the ROLL call to IMS.

**User response:** If necessary, check the reason why your application issues the ROLL call.
**PSSR messages**

These topics provide FABS messages which are issued by the Physical Sequence Sort for Reload (PSSR) utility.

<table>
<thead>
<tr>
<th>Message Code</th>
<th>Message Text</th>
<th>Explanation</th>
<th>System Action</th>
<th>User Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>FABS3501E</td>
<td>LOAD FAILED FOR &quot;xxxxxxxx&quot; IN &quot;yyyy&quot; DATA SET</td>
<td>A LOAD macro was issued for the specified module, but no module could be loaded. If the load module should be loaded from the STEPLIB (or JOBLIB) data set, the data set name is not displayed on the message text.</td>
<td>PSSR issues a user 3501 abend.</td>
<td>Ensure that the indicated member exists in the data set that is specified by the indicated ddname. Correct the error and rerun the job.</td>
</tr>
<tr>
<td>FABS3502E</td>
<td>FAILURE IN DEVTYPE MACRO FOR THE CTL DATA SET</td>
<td>When the DEVTYPE macro was issued for the CTL data set, it generated a return code of 8.</td>
<td>PSSR issues a user 3502 abend.</td>
<td>Ensure that the CTL DD statement is correct. Correct any JCL errors and rerun the job.</td>
</tr>
<tr>
<td>FABS3503E</td>
<td>&quot;xxxxxx&quot; DD DATA SET OPEN FAILED</td>
<td>OPEN processing failed for the data set that is specified by the xxxxxx DD statement.</td>
<td>PSSR issues a user 3503 abend.</td>
<td>Ensure that the xxxxxx DD statement exists, and that it specifies the correct DD parameter. Correct any errors, and rerun the job.</td>
</tr>
<tr>
<td>FABS3504I</td>
<td>DBRC REQUIRED FOR HIGH AVAILABILITY LARGE DATABASE &quot;ddddddddddm&quot;</td>
<td>This message notifies that the database is a HALDB and the PSSR processes it internally with DBRC=Y.</td>
<td>PSSR continues processing.</td>
<td>None. This message is informational.</td>
</tr>
<tr>
<td>FABS3505E</td>
<td>&quot;xxxxxxxx&quot; COULD NOT BE FOUND</td>
<td>A FIND macro was issued for the specified module. No module with that name was in the library that is specified in the IMS2 DD statement.</td>
<td>PSSR issues a user 3505 abend.</td>
<td>Ensure that the correct partitioned data set is specified in your IMS2 DD statement. Correct the error and rerun the job.</td>
</tr>
<tr>
<td>FABS3506E</td>
<td>LOAD MODULE FABSSC00 IS NOT REENTRANT</td>
<td>The FABSSC00 is not a re-entrant module.</td>
<td>PSSR issues a user 3506 abend.</td>
<td>Re-link the FABSSC00 load module with the re-entrant attribute.</td>
</tr>
<tr>
<td>FABS3507E</td>
<td>NONZERO RETURN CODE FROM IDENTIFY</td>
<td>The IDENTIFY macro generated a nonzero return code. When IDENTIFY macro issued for DFSRRC00, it failed.</td>
<td>PSSR issues a user 3507 abend.</td>
<td>Contact IBM Software Support.</td>
</tr>
<tr>
<td>FABS3508E</td>
<td>FAILURE IN LINK MACRO FOR DFSORT MODULE</td>
<td>When the LINK macro was issued for the DFSORT module, it failed.</td>
<td>PSSR issues a user 3508 abend.</td>
<td>Ensure that the DFSORT module is in the system library correctly. Correct the error and rerun the job.</td>
</tr>
<tr>
<td>FABS3509E</td>
<td>PART STATEMENT IS SPECIFIED FOR PSSR OF PDB NON-SUPPORT VERSION</td>
<td>A PART statement for the Partition Data Base (PDB) is specified for PSSR of PDB nonsupport version.</td>
<td>PSSR issues a user 3509 abend.</td>
<td>Check the PSSR CTL statement. Remove the PART statement, and rerun the job.</td>
</tr>
<tr>
<td>FABS3510E</td>
<td>INVALID STATEMENT FOUND IN PSSRIN DATA SET</td>
<td>A PSSRIN data set contains an incorrect statement.</td>
<td>PSSR issues a user 3510 abend.</td>
<td>Check the syntax of the PSSRIN data set. Correct the errors, and rerun the job.</td>
</tr>
</tbody>
</table>
### FABS3511E  INPUT UNLOAD RECORD FORMAT UNMATCHED FOR INPUT DBD

**Explanation:** The database organization of the unload record does not match that of the DBD.

**System action:** PSSR issues a user 3511 abend.

**User response:** Ensure that the unload files and DBD are correct. Correct the error and rerun the job.

### FABS3512E  GETMAIN FAILURE OCCURRED

**Explanation:** A GETMAIN macro was issued, but it failed.

**System action:** PSSR issues a user 3512 abend.

**User response:** Storage space might not be enough for GETMAIN. Check the region size of the PSSR job. Increase the region size in the PSSR JCL, and rerun the job.

### FABS3513E  LISTCAT FAILED FOR DSN=data_set_name

**Explanation:** PSSR attempted to issue a LISTCAT command for the database data set, but it failed.

**System action:** PSSR issues a user 3513 abend.

**User response:** Ensure that input DBDS exists, or that the IDCAMS module is in the system library correctly. Correct the error and rerun the job.

### FABS3514E  AN INCORRECT SORT EXIT FOR THE DATABASE ORGANIZATION

**Explanation:** The specified name of the sort exit module is not applicable for the database organization, or the database organization is not supported by PSSR.

**System action:** PSSR issues a user 3514 abend.

**User response:** Ensure that the database organization is appropriate and whether the sort exit module that is specified on the SYSIN or SORTCTL DD is applicable for the database organization. Correct the error and rerun the job.

### FABS3515E  IMSPLEX NAME EXCEEDS MAXIMUM ALLOWABLE LENGTH

**Explanation:** The IMSplex name that is specified by the IMSPLEX= on the EXEC parameter for FBSMAIN exceeds the maximum allowable length.

**System action:** PSSR issues a user 3515 abend.

**User response:** Correct the error and rerun the job.

### FABS3516E  DBRCGRP ID EXCEEDS MAXIMUM ALLOWABLE LENGTH

**Explanation:** The DBRCGRP name that is specified by DBRCGRP= on the EXEC parameter for FBSMAIN exceeds the maximum allowable length.

**System action:** PSSR issues a user 3516 abend.

**User response:** Correct the error and rerun the job.

### FABS3520I  SORTOUT3 DD STATEMENT IS OMITTED

**Explanation:** PSSR received a return code of 4 from the randomizing routine, and the corresponding root segment and its dependent segments were discarded because the SORTOUT3 DD statement is not defined in your JCL.

**System action:** PSSR continues processing.

**User response:** If you want to keep the segment records that received a return code 4 from the randomizing routine, specify a SORTOUT3 DD statement in your JCL, and rerun the job.

### FABS3521I  DUMMY SPECIFIED FOR SORTOUT3 FILE

**Explanation:** PSSR received a return code 4 from the randomizing routine, and the corresponding root segment and its dependent segments were discarded because the SORTOUT3 DD statement is defined as DUMMY in your JCL.

**System action:** PSSR continues processing.

**User response:** If you want to keep the segment records that received a return code 4 from the randomizing routine, correct the SORTOUT3 DD statement in your JCL, and rerun the job.

### FABS3522I  BLKSIZE OR LRECL OF SORTOUT3 FILE IS ’0’; MAX ALLOWABLE SIZE FOR DEVICE IS USED

**Explanation:** Either BLKSIZE or LRECL is not specified on your SORTOUT3 DD statement.

**System action:** PSSR continues processing, using the maximum record size for the device assigned to the SORTOUT3 DD statement as the BLKSIZE. It uses BLKSIZE-4 as the LRECL.

**User response:** None, unless the program ends because of to an IMS or MVS error. In that case, code both LRECL and BLKSIZE on your DD statement, and rerun the job.

**Tip:** For some DASD devices (such as 3375 or 3380), the default block size and record length result in wasted space. When the maximum record size for the
device is considerably smaller than the maximum track length, it is desirable to code BLKSIZE and LRECL on your DD statement.

**FABS3563W  xxxxxxxxxx SEGMENTS WERE COPIED INTO SORTOUT3 FILE BECAUSE OF RECEIVE RC=4 FROM RANDOMIZER (yyyyyyyyyy root segments)**

Explanation: PSSR received a return code 4 from the randomizing routine. The unloaded database (output) does not contain the corresponding root segment and all of its dependent segments. xxxxxxxxxx indicates the total number of segments that have been copied into SORTOUT3 file, and yyyyyyyyy indicates the number of root segments that received a return code 4 from the randomizing routine.

System action: PSSR continues processing.

User response: If the user intended the segments to be discarded, no response is required. Otherwise, check why the user’s randomizer returned a return code of 4 for those segments and take an appropriate action. Rerun the PSSR job if necessary.

---

**FABS3564W  xxxxxxxxxx SEGMENTS WERE DISCARDED DUE TO RECEIVE RC=4 FROM RANDOMIZER (yyyyyyyyyy ROOT SEGMENTS)**

Explanation: PSSR received a return code 4 from the randomizing routine. The unloaded database (output) does not contain the corresponding root segment and all of its dependent segments. xxxxxxxxxx indicates the total number of segments that have been discarded, and yyyyyyyyy indicates the number of root segments that received a return code 4 from the randomizing routine.

System action: PSSR continues processing.

User response: If the user intended the segments to be discarded, no response is required. Otherwise, check why the user’s randomizer returned a return code of 4 for those segments and take an appropriate action. Rerun the PSSR job if necessary.

---

**FABS3565W  xxxxxxxxxx SEGMENTS WERE COPIED INTO SORTOUTD FILE (yyyyyyyyyy ROOT SEGMENTS)**

Explanation: PSSR detected duplicate records, and has stored them in the SORTOUTD data set. The unloaded database (output) does not contain the corresponding root segment and all of its dependent segments. xxxxxxxxxx indicates the total number of segments that have been stored into the SORTOUTD data set, and yyyyyyyyy indicates the number of root segments that the root keys are duplicated.

System action: PSSR continues processing.

User response: If the user intended the segments to be unloaded, response is required. Otherwise, check why these root keys were duplicated, and take an appropriate action. Rerun the PSSR job if necessary.

---

**FABS3566W  xxxxxxxxxx SEGMENTS WERE DISCARDED (yyyyyyyyyy ROOT SEGMENTS)**

Explanation: PSSR detected duplicate records and discarded them because DUMMY is specified in the SORTOUTD DD. The unloaded database (output) does not contain the corresponding root segment and all of its dependent segments. xxxxxxxxxx indicates the total number of segments that are discarded, and yyyyyyyyyyyyy indicates the number of root segments that the root keys are duplicated.

System action: PSSR continues processing.

User response: If the user intentionally expected that the segments to be discarded, then no response is required. Otherwise, check why these root keys were duplicated and take an appropriate action. Rerun the PSSR job if necessary.

---

**FABS3568E  ERROR RETURN CODE RECEIVED FROM PARTITION SELECTION**

Explanation: When PSSR issues IEXIT or TEXIT call to a partition selection (DFSPSEL), it received a non-zero return code.

System action: PSSR issues a user 3568 abend.

User response: Ensure that the DFSPSEL module or your partition selection exit is correct. Correct the error and rerun the job.

---

**FABS3569E  PARTITION SELECTION EXIT NOT SELECTED ALL PARTITIONS**

Explanation: Some partitions are not selected during partition selection by the partition selection exit.

System action: PSSR issues a user 3569 abend.

User response: Ensure that when the partition selection exit is called by the NEXT function, it establishes partition chaining with all of partition ID. Correct the error and rerun the job.

---

**FABS3570E  DBD NAME ON CONTROL STATEMENT COULD NOT BE FOUND IN “IMS” DATA SET**

Explanation: A FIND macro was issued for the DBD name that the FABSRT15 control statement specifies. No module with that name is in the library on your IMS DD statement.

System action: FABSRT15 issues a user 3570 abend.

User response: Check the FABSRT15 control statement (in the CTL DD statement) for spelling errors in the
FABS3571E  INVALID VALUE SPECIFIED ON DB STATEMENT AT COLUMN xx

Explanation: An incorrect value is specified at column xx for the DB statement. yyyyyyyyy indicates the specified incorrect value.

System action: PSSR discards the incorrect value and treats it as if a blank is specified on that field. It might cause the Record Size Distribution report to be printed unexpectedly.

User response: If the action of PSSR is not acceptable, check the PSSR control statement (in the CTL DD statement) for an error in the indicated column, and rerun the job.

FABS3572W  INVALID OPTION SPECIFIED ON DB STATEMENT AT COLUMN xx

Explanation: An incorrect value is specified at column xx of the DB statement (in the CTL DD statement). PSSR discards the incorrect value and treats it as if a blank is specified on that field.

System action: PSSR continues processing as if a blank is specified on that column.

User response: If the action of PSSR is not acceptable, check the PSSR control statement (in the CTL DD statement) for an error in the indicated column, and rerun the job.

FABS3573E  INVALID VALUE SPECIFIED ON PART STATEMENT FOR partname AT COLUMN xx

Explanation: An incorrect value is specified at column xx of the PART statement (in the CTL DD statement) partname. yyyyyyyyy indicates the specified value.

System action: PSSR discards the incorrect value and treats it as if a blank is specified on that field.

User response: If the action of PSSR is not acceptable, check the specification of the PART statement. Specify a correct value for the parameter and rerun the job.

FABS3574W  INVALID VALUE SPECIFIED ON DB STATEMENT

Explanation: An incorrect value is specified on the DB statement (in the CTL DD statement). PSSR discards the incorrect value.

System action: PSSR continues processing.

User response: None.

FABS3575E  PARTITION SELECTION EXIT NAME WAS NOT SPECIFIED

Explanation: This message indicates a PSSR CTL statement error. "S" is specified in column 72 on DB statement (in the CTL DD statement), but the name of the partition selection exit is not specified.

System action: PSSR issues a user 3575 abend.

User response: Check the CTL DD statement. Correct the errors, and rerun the job.

FABS3576E  PARTITION partname IS NOT FOUND IN THE DB: dbdname

Explanation: The partition partname specified in column 2 of the PART statement is not found in the DB dbdname.

System action: PSSR discards the incorrect option that is specified in column 72 on the PART statement (in the CTL DD statement), but the name of the partition selection string or high key was not specified correctly.

System action: PSSR issues a user 3576 abend.

User response: Check the PART statement. Remove "+" from column 72 in the PART statement, or specify the partition selection string or high key.

FABS3577W  INVALID VALUE SPECIFIED ON PART STATEMENT FOR PHIDAM

Explanation: An incorrect value is specified on the PART statement (in the CTL DD statement). PSSR ignores the incorrect value.

System action: PSSR continues processing.

User response: None.

FABS3578E  PARTITION SELECTION STRING OR HIGH KEY WAS NOT SPECIFIED

Explanation: "+" is specified in column 72 in the PART statement (in the CTL DD statement), but, the partition selection string or high key was not specified correctly.

System action: PSSR issues a user 3578 abend.

User response: Check the PART statement. Remove "+" from column 72 in the PART statement, or specify the partition selection string or high key.

FABS3579W  INVALID OPTION SPECIFIED ON PART STATEMENT AT COLUMN xx

Explanation: An incorrect value is specified on the DB statement (in the control statement (in the CTL DD statement)). PSSR discards the incorrect option that is specified in column xx and treats it as if a blank is specified in that field.

System action: PSSR continues processing as if a blank is specified in column xx.

User response: If the action of PSSR is not acceptable, check the PSSR control statement (in the CTL DD statement) for an error in column xx. Correct the value
in the column and rerun the job.

**FABS3580E**  “CTL” FILE CONTAINS NO INPUT CONTROL STATEMENTS

Explanation: The data set that is specified on your CTL DD statement is empty.

System action: FABSRT15 issues a user 3580 abend.

User response: Check the CTL DD statement for JCL errors. Correct the error and rerun the job.

**FABS3581E**  FAILURE IN DEVTYPE MACRO FOR THE SORTOUT3 FILE

Explanation: The DEVTYPE macro generated a return code 8 when it was issued for the SORTOUT3 data set.

System action: PSSR issues a user 3581 abend.

User response: Check the SORTOUT3 DD statement for JCL errors. Correct the error and rerun the job.

**FABS3582E**  SORTOUT3 DD DATA SET OPEN FAILED

Explanation: OPEN processing failed for the data set that is specified by the SORTOUT3 DD statement.

System action: PSSR issues a user 3582 abend.

User response: Ensure that the SORTOUT3 DD statement exists, and that it specifies the correct DD parameter. Correct any errors, and rerun the job.

**FABS3583E**  RANDOMIZING ROUTINE RETURNED WITH RC=8

Explanation: PSSR received a return code 8 from the randomizing routine.

System action: PSSR issues a user 3583 abend.

User response: None.

**FABS3584E**  SORTOUT3 DD DATA SET CLOSE FAILED

Explanation: CLOSE processing failed for the data set that is specified by the SORTOUT3 DD statement.

System action: PSSR issues a user 3584 abend.

User response: Check the SORTOUT3 DD statement.

**FABS3585E**  INVALID LENGTH OF PARTITION SELECTION STRING OR HIGH KEY

Explanation: The partition selection string or the high key in the PART statement is not equal to the key.

System action: PSSR issues a user 3585 abend.

User response: Check the PART statement from the second line on, in the DBD. Correct the errors, and rerun the job.

**FABS3586E**  INVALID VALUE SPECIFIED FOR PARTITION SELECTION STRING OR HIGH KEY

Explanation: The partition selection string or the high key in the PART statement is incorrect.

System action: PSSR issues a user 3586 abend.

User response: Check the PART statement from the second line on. Correct the errors and rerun the job.

**FABS3587E**  DUPLICATE VALUE SPECIFIED FOR PARTITION SELECTION STRING OR HIGH KEY

Explanation: The same value is specified for two partitions as the partition selection string or the partition high key.

System action: PSSR issues a user 3587 abend.

User response: Check the PART statement (in the CTL DD statement) and the actual HALDB definitions for the partition selection string or the partition high key. Correct the error and rerun the job.

**FABS3590E**  INVALID FIRST RECORD ON "HDHEADER" FILE

Explanation: The second byte in the first record in the HDHEADER data set contained a value not equal to X'00'.

System action: FABSRT35 issues a user 3590 abend.

User response: Check the HDHEADER DD statement for JCL errors. Correct the error and rerun the job.

**FABS3591E**  UNEXPECTED END-OF-DATA ON "HDHEADER" FILE

Explanation: An end-of-data is reached on the HDHEADER data set. It contained less than two records.

System action: FABSRT35 issues a user 3591 abend.

User response: Check the HDHEADER DD statement for JCL errors. Correct the error and rerun the job.

**FABS3592E**  DO NOT USE A LARGE FORMAT D/S: dsname

Explanation: The *dsname* data set is a large format data set that is supported by z/OS Version 1 Release 7 and later. IMS HP Load cannot process large format data sets in z/OS Version 1 Release 6 or that of prior versions.

System action: PSSR issues a user 3588 abend and a dump.
<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FABS3593E</td>
<td>GETDSAB FAILED FOR ddname</td>
</tr>
<tr>
<td>Explanation: The GETDSAB macro failed for DD name ddname.</td>
<td></td>
</tr>
<tr>
<td>System action: PSSR issues a user 3589 abend and a dump.</td>
<td></td>
</tr>
<tr>
<td>User response: This problem is likely to be an internal system error. Contact IBM Software Support.</td>
<td></td>
</tr>
<tr>
<td>FABS3594E</td>
<td>SWAREQ FAILED FOR ddname</td>
</tr>
<tr>
<td>Explanation: The SWAREQ macro failed for DD name ddname.</td>
<td></td>
</tr>
<tr>
<td>System action: PSSR issues a user 3590 abend and a dump.</td>
<td></td>
</tr>
<tr>
<td>User response: This problem is likely to be an internal system error. Contact IBM Software Support.</td>
<td></td>
</tr>
<tr>
<td>FABS3595E</td>
<td>OBTAIN FAILED FOR ddname</td>
</tr>
<tr>
<td>Explanation: The OBTAIN macro failed for DD name ddname.</td>
<td></td>
</tr>
<tr>
<td>System action: PSSR issues a user 3591 abend and a dump.</td>
<td></td>
</tr>
<tr>
<td>User response: This problem is likely to be an internal system error. Contact IBM Software Support.</td>
<td></td>
</tr>
<tr>
<td>FABS3596E</td>
<td>ERROR RETURN FROM IGWASYS (RC=return_code, RSN=reason_code)</td>
</tr>
<tr>
<td>Explanation: IGWASYS returned an error return code. Here, RC shows the return code and RSN shows the reason code.</td>
<td></td>
</tr>
<tr>
<td>System action: PSSR issues a user 3592 abend and a dump.</td>
<td></td>
</tr>
<tr>
<td>User response: See DFSMSdfp Advanced Services and check the meaning of the return code and reason code.</td>
<td></td>
</tr>
<tr>
<td>FABS3600E</td>
<td>KEY BLOCK NUMBERS ARE NOT IN ASCENDING SEQUENCE</td>
</tr>
<tr>
<td>Explanation: DFSORT (output phase) returned a record that was out of sequence. The block number (that was added by FABSRT15) was out of sequence.</td>
<td></td>
</tr>
<tr>
<td>System action: FABSRT35 issues a user 3600 abend.</td>
<td></td>
</tr>
<tr>
<td>User response: Check the SY SIN DD statement for JCL or control statement errors. Check for a system failure that is external to PSSR. Correct the error and rerun the job.</td>
<td></td>
</tr>
<tr>
<td>FABS3610E</td>
<td>RBA OF THE KEY BLOCK IS LESS THAN THE RBA OF THE CURRENT BEGINNING BLOCK</td>
</tr>
<tr>
<td>Explanation: Data got out of sequence while creating the PR9X data set.</td>
<td></td>
</tr>
<tr>
<td>System action: FABSRT35 issues a user 3610 abend.</td>
<td></td>
</tr>
<tr>
<td>User response: Check for a system failure that is external to PSSR. Correct the error and rerun the job.</td>
<td></td>
</tr>
<tr>
<td>FABS3611E</td>
<td>HALDB dbdname IS DEFINED AS OSAM8G</td>
</tr>
<tr>
<td>Explanation: The indicated HALDB is defined as OSAM8G in the RECON data set. The maximum capacity of OSAM data sets is 8 GB of data.</td>
<td></td>
</tr>
<tr>
<td>System action: PSSR continues processing.</td>
<td></td>
</tr>
<tr>
<td>User response: None. This message is informational.</td>
<td></td>
</tr>
<tr>
<td>FABS3617E</td>
<td>KEY RAP NUMBER IS ZERO</td>
</tr>
<tr>
<td>Explanation: DFSORT (output phase) returned a record in which the RAP number (that was added by FABSRT15) is zero.</td>
<td></td>
</tr>
<tr>
<td>System action: FABSRT35 issues a user 3617 abend.</td>
<td></td>
</tr>
<tr>
<td>User response: Check the SY SIN DD statement for JCL or control statement errors. Check for a system failure that is external to PSSR. Correct the error and rerun the job.</td>
<td></td>
</tr>
<tr>
<td>FABS3620E</td>
<td>RAP NUMBER IS GREATER THAN THE MAXIMUM NUMBER OF RAPS IN A BLOCK</td>
</tr>
<tr>
<td>Explanation: DFSORT returned, in the output phase, a record in which the RAP number added by PSSR was greater than the maximum number of RAPs in a block.</td>
<td></td>
</tr>
<tr>
<td>System action: FABSRT35 issues a user 3620 abend.</td>
<td></td>
</tr>
<tr>
<td>User response: Check the HDHEADER DD statement for JCL errors. Check for a system failure that is external to PSSR. Correct the error and rerun the job.</td>
<td></td>
</tr>
<tr>
<td>FABS3624I</td>
<td>DUMMY SPECIFIED FOR THE SOROUTD FILE</td>
</tr>
<tr>
<td>Explanation: The DD statement for SOROUTD is specified as DUMMY. If duplicated root key segments are detected by PSSR, the database segment records are discarded.</td>
<td></td>
</tr>
<tr>
<td>System action: PSSR continues processing.</td>
<td></td>
</tr>
<tr>
<td>User response: If you want to keep the duplicate root key segment records, correct the SOROUTD DD statement in your JCL and rerun the job.</td>
<td></td>
</tr>
</tbody>
</table>
FABS3625E  ddname DD STATEMENT IS MISSING
Explanation: The DD statement for ddname is not specified on your JCL.
System action: PSSR issues a user 3625 abend.
User response: Specify the DD statement and rerun the job.

FABS3626E  FAILURE IN DEVTYPE MACRO FOR ddname FILE
Explanation: The DEVTYPE macro returned RC=08 for the DD ddname.
System action: PSSR issues a user 3626 abend.
User response: Check if the DD statement for ddname is correct. Correct your JCL and rerun the job.

FABS3627E  DUMMY SPECIFIED FOR THE ddname FILE
Explanation: The DD statement for ddname is specified as DUMMY. For the SORTOUT2 DD, this message is issued only when the byte limit is specified for the partitioned database.
System action: PSSR issues a user 3627 abend.
User response: SORTOUT2 DD must be specified as non-DUMMY if you specify the byte limit for the partition DD statement:
• Specify a data set name for the DD if you want the segment records for the partition identified by the ddname to be put into the data set.
• Remove the DD statement if you want the segment records for the partition to be put into the SORTOUT data set.

FABS3629I  BLKSIZE OR LRECL OF ddname FILE IS '0'; MAX ALLOWABLE SIZE FOR DEVICE IS USED
Explanation: BLKSIZE or LRECL is not coded on the DD statement for ddname.
System action: PSSR uses the maximum record size for the device assigned to the DD as its BLKSIZE. It uses (BLKSIZE - 4) as the LRECL. Processing continues.
User response: If PSSR ends normally and your subsequent IMS HD Reorganization Reload utility steps work, no response is needed. Otherwise, make sure that the SORTIN data set and the data set that is specified on the ddname DD statement have the same DCB values. Rerun the job.

FABS3630E  DBD ERROR - ROOT SEGMENT NOT FOUND IN DBD SEGTAB
Explanation: PSSR attempted to process a DBD that was invalid.
System action: PSSR issues a user 3630 abend.
User response: Check the FABSRT15 control statement (in the CTL DD statement) for spelling errors in the ddname (column 1) field. Ensure that the correct partitioned data set is specified on your IMS DD statement. Correct the error and rerun the job.

FABS3640E  UNLOADED DATABASE (INPUT) CONTAINS A RECORD OF WHICH TYPE IS NOT RECOGNIZABLE
Explanation: The data set that is specified on your SORTIN DD statement contains an invalid record. The PSSR can accept an unloaded file that is created in either the *HD or the *SH format. The CS format is applicable when the input DBD is defined as a non-HALDB.
System action: PSSR issues a user 3640 abend.
User response: Check your SORTIN DD statement for JCL errors. Correct the PSSR JCL. Rerun DFSURGU0 to recreate the SORTIN data set, if necessary. Then, rerun the PSSR job.
If no JCL errors are found, then the data set has probably been corrupted.

FABS3641E  DUPLICATE KEY CHECKING CANNOT BE SPECIFIED TO THE COMPRESSED ROOT KEY
Explanation: The SORTOUTD DD statement is specified in the PSSR JCL, but the data set that is specified on the SORTIN DD statement contains a compressed root key. PSSR cannot check the duplicate root keys when the root keys are compressed.
System action: PSSR issues a user 3641 abend.
User response: If the duplicate root keys checking is not necessary, remove the SORTOUTD from the PSSR JCL. If the duplicate root keys checking is necessary, submit the reload job again with decompressing, and rerun the PSSR job with the decompressed unloaded file.

FABS3642E  A SEGMENT NAME: mmmmmmm IN UNLOAD DATA SET IS NOT FOUND IN DBD
Explanation: PSSR is unable to find a segment name mmmmmmm in the DBD that matches one from the unloaded data set.
System action: PSSR issues a user 3642 abend and dump.
User response: Ensure that you are running with the correct DBD. Correct the error and rerun the job.

FABS3650E UNLOADED DATABASE (INPUT) CONTAINS A RECORD WITH AN INVALID SEGMENT CODE

Explanation: The data set that is specified on your SORTIN DD statement contains a record whose first byte is larger than the largest segment code that is in your DBD.

System action: FABSRT15 issues a user 3650 abend.

User response: Check your SORTIN DD statement for JCL errors. Ensure that you are using the correct data set and the correct DBD (check the IMS DD statement). Correct the PSSR JCL. If necessary, rerun DFSURGU0 to recreate the SORTIN data set. Rerun the PSSR job.

If no JCL errors are found, then the data set might be corrupted.

FABS3651E NO HEADER RECORD IN SORTIN

Explanation: No header record is found in the SORTIN data set.

System action: PSSR issues a user 3651 abend.

User response: Check that the data set name specified on the SORTIN DD statement is correct. If it is correct, check that the content of the data set is the one you have created using the IMS HD Reorganization Unload utility or an equivalent program, such as the IMS High Performance Unload for z/OS unload utilities. Correct the error and rerun the job.

FABS3652E NO TRAILER RECORD IN SORTIN

Explanation: No trailer record is found in the SORTIN data set.

System action: PSSR issues a user 3652 abend.

User response: Check if the data set name specified on the SORTIN DD statement is correct. If it is correct, check if the content of the data set is the one you have created using the IMS HD Reorganization Unload utility or an equivalent program, such as the IMS High Performance Unload for z/OS unload utilities. Correct the error and rerun the job.

FABS3653E SEGMENT COUNT IN TRAILER IS NOT EQUAL TO REAL SEGMENT COUNT

Explanation: The segment count stored in the trailer record in the SORTIN data set is not equal to the real segment count.

System action: PSSR issues a user 3653 abend.

User response: Check if the data set name specified on the SORTIN DD statement is correct. If it is correct, check if the content of the data set is the one you have created using the IMS HD Reorganization Unload utility or an equivalent program, such as the IMS High Performance Unload for z/OS unload utilities. Correct the error and rerun the job.

FABS3654E UNABLE TO PROCESS THE UNLOAD FILE DUE TO INVALID STAT RECORD

Explanation: The statistics records in the unload data set are not recognizable by PSSR.

System action: PSSR issues a user 3654 abend.

User response: Contact IBM Software Support.

FABS3660I SORTOUT4 DD STATEMENT IS OMITTED

Explanation: PSSR received a nonzero return code from the partition selection, and the root segment and its dependent segments were discarded because the SORTOUT4 DD statement is not defined in your JCL.

System action: PSSR continues processing.

User response: If you want to keep the segment records that received a nonzero return code from the partition selection, specify a SORTOUT4 DD statement in your JCL and rerun the job.

FABS3661I DUMMY SPECIFIED FOR THE SORTOUT4 FILE

Explanation: PSSR received a nonzero return code from the partition selection, and the corresponding root segment and its dependent segments were discarded because the SORTOUT4 DD statement is defined as DUMMY in your JCL.

System action: PSSR continues processing.

User response: If you want to keep the segment records that received a nonzero return code from the partition selection, correct the SORTOUT4 DD statement in your JCL and rerun the job.

FABS3662I BLKSIZE OR LRECL OF SORTOUT4 FILE IS '0'; MAX ALLOWABLE SIZE FOR DEVICE IS USED

Explanation: Either BLKSIZE or LRECL is not specified on your SORTOUT4 DD statement.

System action: PSSR continues processing, using the maximum record size for the device assigned to the SORTOUT4 DD statement as BLKSIZE. It uses BLKSIZE minus 4 as LRECL.

User response: If the program ends because of an IMS or MVS error code both LRECL and BLKSIZE on your DD statement, and rerun the job. Otherwise, none.
Tip: For some DASD devices (such as 3375 or 3380), the default block size and record length result in wasted space. When the maximum record size for the device is considerably smaller than the maximum track length, it is desirable to code BLKSIZE and LRECL on your DD statement.

**FABS3663W**

xxxxxxx SEGMENTS WERE COPIED INTO SORTOUT4 FILE BECAUSE PARTITION SELECTION FAILURE

(yyyyyyyyyy ROOT SEGMENTS)

Explanation: PSSR received an error return code from the partition selection exit or error status because of outside the range of partition high key selection. The unloaded database (output) does not contain the corresponding root segment and all of its dependent segments. xxxxxxxx is the total number of segments that have been copied to SORTOUT4 file, and yyyy yyyy is the number of root segments that received the error return codes from the partition selection.

System action: PSSR continues processing.

User response: Check why the partition selection for those segments failed, and take appropriate action. Rerun the PSSR job, if necessary.

**FABS3664W**

xxxxxxx SEGMENTS WERE DISCARDED BECAUSE TO PARTITION SELECTION FAILURE

(yyyyyyyyyy ROOT SEGMENTS)

Explanation: PSSR received an error return code from the partition selection exit or error status because of outside the range of partition high key selection. The unloaded database (output) does not contain the corresponding root segment and all of its dependent segments. xxxxxxxx is the total number of segments that have been discarded, and yyyy yyyy is the number of root segments that received error return code from the partition selection.

System action: PSSR continues processing.

User response: Check why the partition selection for the segments failed, and take appropriate action. Rerun the PSSR job, if necessary.

**FABS3665E**

FAILURE IN DEVTYPE MACRO FOR SORTOUT4 FILE

Explanation: The DEVTYPE macro returned the nonzero return code for the SORTOUT4.

System action: PSSR issues a user 3665 abend.

User response: Ensure that the DD statement for SORTOUT4 is correct. Rerun the job.

**FABS3666E**

SORTOUT4 DD DATA SET OPEN FAILED

Explanation: OPEN processing failed for the data set that is specified by the SORTOUT4 DD statement.

System action: PSSR issues a user 3666 abend.

User response: Check the SORTOUT4 DD statement for JCL errors. Ensure that the SORTOUT4 DD statement exists, and that it specifies the correct DD parameter. Correct any errors, and rerun the job.

**FABS3667E**

SORTOUT4 DD DATA SET CLOSE FAILED

Explanation: CLOSE processing failed for the data set that is specified by the SORTOUT4 DD statement.

System action: PSSR issues a user 3667 abend.

User response: Check the SORTOUT4 DD statement.

**FABS3670E**

RANDOMIZING ROUTINE COULD NOT BE FOUND IN “IMS2” FILE

Explanation: A FIND macro was issued for the HDAM randomizing routine that is specified either on your FABSR15 control statement or in your DBD load module. No module with that name is in the library on your IMS2 DD statement.

System action: FABSR15 issues a user 3670 abend.

User response: Check the FABSR15 control statement (in the CTL DD statement) for spelling errors in the mod (column 11) field. Ensure that the correct partitioned data set is specified on your IMS2 DD statement. Correct the error and rerun the job.

**FABS3671E**

PARTITION SELECTION EXIT COULD NOT BE FOUND IN IMS2 DD

Explanation: A FIND macro was issued for the partition selection exit routine specified in your PSSR control statement. No module with that name is found in the library on your IMS2 DD statement.

System action: PSSR issues a user 3671 abend.

User response: Check the PSSR control statement (in the CTL DD statement) for spelling errors, or make sure the correct data set is specified on your IMS2 DD statement. Correct the error and rerun the job.

**FABS3680W**

RAA SIZE IS LARGER THAN OR EQUAL TO 4 GIGABYTES

Explanation: PSSR attempted to process a PSSR control statement that contained inconsistent values in the blk or rbn field. The product of these two values was larger than or equal to 4 GB for OSAM data sets of even block size.

System action: Processing continues.
FABS3681E • FABS3714E

User response: If your subsequent reload step allows to process an OSAM database of size larger than or equal to 4 GB, no response is needed. If not, reduce the size of the root addressable area.

FABS3681E  RAA SIZE IS LARGER THAN OR EQUAL TO 4GB FOR THE PARTITION: ddname

Explanation: The size of the root addressable area, which is calculated from the rbn and blksz values that are specified on the PART statement for the partition ddname or specified in the DBD, is larger than or equal to 4 GB. For the partitioned database, the size of the root addressable area of a partition must be less than 4 GB.

System action: PSSR issues a user 3681 abend.

User response: Check the rbn or blksz value specified on the PART statement. Correct the error and rerun the job.

FABS3699E  "BLKSZIN" OR "NBLKSGIN" (ON CONTROL STATEMENT) IS TOO LARGE

Explanation: PSSR attempted to process a FABSRT15 control statement that contained inconsistent values in the blk or rbn fields. The product of these two values must be less than 4 GB for VSAM and OSAM database data set. For OSAM database data set of even block size in IMS Version 5 Release 1 with APAR PN82671 applied or later, the maximum data set size is 8 GB minus 1.

System action: PSSR issues a user 3699 abend.

User response: Check the FABSRT15 control statement (in the CTL DD statement) for errors in the blk or rbn (column 34 or 20) fields. Correct the error and rerun the job.

FABS3700E  KEY POSITION OR LENGTH SPECIFIED ON CTL IS INVALID

Explanation: PSSR attempted to process a PSSR control statement that contains inconsistent values in the kstr or klen fields. The key field which is determined by kstr or klen must be inside the root segment.

System action: PSSR issues a user 3700 abend.

User response: Check the PSSR control statement (in the CTL DD statement) for errors in the kstr or klen (column 41 or 46) field. Correct any errors and rerun the job.

FABS3710E  DBD-OVERRIDE FLAG MUST BE SPECIFIED FOR PARTITIONED DB

Explanation: A value that overrides a (default) DBD parameter value is specified, although the "*" sign is not specified at column 72 of the DB statement.

System action: PSSR issues a user 3710 abend.

User response: If you want to override the parameter value, specify the "*" sign on column 72 of the DB statement. If you do not want to override the parameter values, remove all overriding values. Then, rerun the PSSR job.

FABS3711E  'RBN' ON DB STATEMENT MUST BE BLANK FOR PARTITIONED DB

Explanation: The rbn field of the DB statement is not blank although the database you are processing is a partitioned database. The rbn field on the DB statement cannot be specified for a partitioned database.

System action: PSSR issues a user 3711 abend.

User response: If you want to change the rbn values for the partitioned database, specify the new values on each PART statement.

FABS3712E  PARTITION ddname IS NOT FOUND IN THE DBD: dbdname

Explanation: The DD name ddname specified on column 11 of a PART statement is not found in any PART definitions in the DBD dbdname.

System action: PSSR issues a user 3712 abend.

User response: Ensure that the DD name specified on the PART statement for your PSSR job is correct. Correct any errors and rerun the job.

FABS3713E  THE NUMBER OF TOTAL BLOCKS EXCEEDS THE MAXIMUM

Explanation: The total number of blocks in the root addressable area of all partitions exceeds the allowable maximum. The maximum is 16,777,215.

System action: PSSR issues a user 3713 abend.

User response: Reduce the rbn value on each PART statement of the partition specified in the CTL data set so that the total RBN value throughout the database does not exceed the allowable maximum.

FABS3714E  'BLKSZ' ON PART STATEMENT IS NOT AN EVEN NUMBER

Explanation: The block size that is specified on a PART control statement is not an even number. For the partitioned database, the block size or CI size of each partition data set must be even.

System action: PSSR issues a user 3714 abend.
User response: Specify an even block size and rerun the job.

---

**FABS3715E** PART STATEMENT IS SPECIFIED FOR NON-PARTITIONED DB

Explanation: A PART statement is specified for a nonpartitioned database.

System action: PSSR issues a user 3715 abend.

User response: Check whether the specified DBD is for a nonpartitioned database.

If you specified the incorrect DBD, specify the correct one and rerun the job. If you specified the correct DBD, remove the PART statement and rerun the job.

---

**FABS3716E** DUPLICATED PART STATEMENTS ARE FOUND

Explanation: The PART statement for a partition is specified more than once. Only one PART statement is allowed for each partition.

System action: PSSR issues a user 3716 abend.

User response: Correct the error and rerun the job.

---

**FABS3717E** DUPLICATED DB STATEMENTS ARE FOUND

Explanation: The DB statement is specified more than once. Only one DB statement is allowed.

System action: PSSR issues a user 3717 abend.

User response: Correct the error and rerun the job.

---

**FABS3718E** INVALID STATEMENT IS SPECIFIED IN CTL DATA SET

Explanation: An incorrect statement is specified in the CTL data set. Only the DB statement and the PART statements are allowed.

System action: PSSR issues a user 3718 abend.

User response: Correct the error and rerun the job.

---

**FABS3719E** DB STATEMENT IS NOT SPECIFIED IN CTL DATA SET

Explanation: No DB statement is specified in the CTL data set. The DB statement is required.

System action: PSSR issues a user 3719 abend.

User response: Specify a DB statement and rerun the job.

---

**FABS3720E** BLKSZ ON DB STATEMENT MUST BE BLANK FOR PARTITIONED DB

Explanation: The blksz field of the DB statement is not blank although the database you are processing is a partitioned database. The blksz field on the DB statement must not be specified for a partitioned database.

System action: PSSR issues a user 3720 abend.

User response: If you want to change the blksz values for a partitioned database, specify the new values on each PART statement.

---

**FABS3730E** INVALID PARTITION NUMBER 'xxxxxxxx' RETURNED FROM RANDOMIZER: DBFyyyyyy

Explanation: An incorrect partition number is returned in Register 1 from the randomizer DBFyyyyyy. xxxxxxxx shows the content of Register 1 in hexadecimal.

System action: PSSR issues a user 3730 abend.

User response: Check the logic of your randomizer. For the partitioned database, the randomizing routine with the load module name that begins with "DBF" is assumed to have the DEDB interface.

If there is an error in your randomizer logic or in the naming of the randomizer, correct the error and rerun the job.

---

**FABS3731E** INVALID RAP NUMBER 'xxxxxxxx' RETURNED FROM RANDOMIZER: DBFyyyyyy

Explanation: An incorrect relative RAP number is returned in Register 0 from the randomizer DBFyyyyyy. xxxxxxxx shows the content of Register 0 in hexadecimal.

System action: PSSR issues a user 3731 abend.

User response: Check the logic of your randomizer. For the partitioned database, the randomizing routine with the load module name that begins with "DBF" is assumed to have the DEDB interface.

If there is an error in your randomizer logic or in the naming of the randomizer, correct the error and rerun the job.

---

**FABS3740I** USEREXIT STATEMENT IS IGNORED BECAUSE OF NON-IMS ENVIRONMENT

Explanation: USEREXIT in the PSSRIN data set is ignored because the environment is not IMS. The PSSRIN data set can be specified only in the IMS environment.

System action: PSSR continues processing.

---

Chapter 19. Messages and codes  307
User response: None. This message is informational.

FABS3741E  DUPLICATED USEREXIT STATEMENTS ARE FOUND
Explanation: The USEREXIT statement is specified more than once in the PSSRIN control statement. Only one USEREXIT statement is allowed.
System action: PSSR issues a user 3741 abend.
User response: Correct the errors and rerun the job.

FABS3742E  USER EXIT ROUTINE "uuuuuuuu" COULD NOT BE FOUND IN IMS2 DD
Explanation: A FIND macro was issued for the user exit routine that is specified on the PSSRIN control statement. No module with that name is found in the library on your IMS2 DD statement.
System action: PSSR issues a user 3742 abend.
User response: Check the PSSRIN control statement (in the PSSRIN DD statement) for spelling errors, and ensure that the correct partitioned data set is specified on your IMS2 DD statement. Correct any errors and rerun the job.

FABS3743E  INVALID RETURN CODE FROM USER EXIT ROUTINE, RETURN (CODE=xxx or RETURN CODE=GREATERTHAN 999)
Explanation: A return code xxx, or greater than 999, was returned by the user exit routine.
System action: PSSR issues a user 3743 abend and a dump.
User response: Correct the errors and rerun the job.

FABS3744E  TERMINATION REQUESTED BY USER EXIT ROUTINE, RETURN CODE=xxx
Explanation: The job step terminated and returned a return code of xxx.
System action: The PSSR job ends immediately with a return code of xxx.
User response: Check the return code of the user exit routine. If the return code is not acceptable, correct the errors and rerun the job.

FABS3745E  INVALID SEGMENT LENGTH RETURNED FROM USER EXIT ROUTINE
Explanation: A user exit routine changed the length of a database segment to a value greater than the allowed maximum segment length.

FABS3746E  INVALID LANGUAGE ENVIRONMENT IS SPECIFIED ON USEREXIT STATEMENT
Explanation: An incorrect value is specified for the language of the user exit in the PSSRIN data set.
System action: PSSR issues a user 3746 abend.
User response: Check the PSSRIN data set. Correct any errors and rerun the job.

FABS3750I  DATXEXIT=YES IS IGNORED BECAUSE OF NON-IMS ENVIRONMENT
Explanation: DATXEXIT=YES in the PSSR data set is ignored because the environment is not IMS. The DATXEXIT=YES can be specified only in IMS environments.
System action: PSSR continues processing as DATXEXIT=NO.
User response: None. This message is informational.

FABS3752E  INVALID DATXEXIT STATEMENT SPECIFIED
Explanation: The DATXEXIT statement must be specified as either YES or NO.
System action: PSSR issues a user 3752 abend.
User response: Correct the error and rerun the job.

FABS3753E  LOAD FAILED FOR DFSDBUX1 ROUTINE
Explanation: While processing a database that require data conversion by the DATXEXIT=YES parameter in a DBD, PSSR attempted to load the DFSDBUX1 exit routine, but the load module is not found in the STEPLIB.
System action: PSSR issues a user 3753 abend and a dump.
User response: Ensure that the correct DFSDBUX1 module exists in the STEPLIB. Correct any errors and rerun the job.

FABS3754E  NONZERO RETURN CODE FROM DFSDBUX1 ROUTINE RC=return_code
Explanation: Return code return_code was returned by the DFSDBUX1 exit routine.
System action: PSSR issues a user 3754 abend and a dump.
User response: Correct the errors and rerun the job.

FABS3755E  SPECIFIED DATABASE "ddddd" IS NOT A TARGET FOR CONVERSION

Explanation: While processing the database "ddddd" requiring data conversion by the DATXEXIT=YES parameter in a DBD, PSSR called the DFSDBUX1 exit routine and found that none of the segments required conversion.

System action: PSSR issues a user 3755 abend and a dump.

User response: Ensure that definition of translation table is correct or the correct translation table module exists in STEPLIB. Correct any errors and rerun the job.

FABS3756I  DBDGEN REQUIRED FOR DATABASE "ddddd" TO SET DATXEXIT INDICATOR

Explanation: PSSR called the DFSDBUX1 exit routine and found that some of the database segments required conversion, but the DATXEXIT=YES parameter was not specified in the DBD "ddddd". This message notifies you that a DBDGEN with DATXEXIT=YES specified needs to be done for this database.

System action: PSSR continues processing as specified.

User response: Notify the database administrator that a DBDGEN is required for this database.

FABS3761E  INVALID SEGMENT LENGTH RETURNED FROM COMPRESSION ROUTINE

Explanation: The database segment length that is returned from the segment edit/compression exit routine is larger than the allowed maximum value.

System action: PSSR issues a user 3761 abend and a dump.

User response: Check and correct the DBD definition or the segment edit/compression exit routine, and rerun the job.

FABS3762E  CEEPIPI ERROR OCCURRED.

Explanation: PSSR called CEEPIPI for processing the user exit routine, specified to run in Language Environment (LE), but received a nonzero return code.

System action: PSSR issues a user 3762 abend and a dump.

User response: When function shows INIT, the function of CEEPIPI is init_sub or identify_entry. When function shows CALLSUB, the function of CEEPIPI is call_sub. For the return codes, see the topic about using preinitialization services in the Language Environment Programming Guide (SA22-7561). Correct the error and rerun the job.
Bitmap Resetter messages

These topics provide HPSB messages which are issued by the Bitmap Resetter utility.

**HPSB0100E** - NO DBRC=Y OR DBRC=N IN PARAMETER STRING

**Explanation:** Incorrect DBRC parameter is specified. Only Y or N is valid.

**System action:** Processing stops with a return code of 8.

**User response:** Correct the error and rerun the job.

**HPSB0101E** - INCORRECT DBDNAME IN PARAMETER STRING

**Explanation:** Incorrect DBDNAME is specified.

**System action:** Processing stops with a return code of 8.

**User response:** Correct the error and rerun the job.

**HPSB0102E** - INCORRECT NUMERIC IN BMSETDSG CONTROL STATEMENT:

**Explanation:** Incorrect value is specified for the BMSETDSG control statement or the control statement is not specified. You can specify ‘*', 'NO', or a numerical character in the range of 16 - 32000. Reason shows the cause of the error as follows:

- *LESS THAN 16
- *GREATER THAN 32000
- *NEGATIVE MISSING

**System action:** Processing stops with return code 8.

**User response:** Correct the error and rerun the job.

**HPSB0103E** - GETMAIN FAILED FOR GLOBAL BLOCK. TERMINATING

**Explanation:** GETMAIN failed while the Bitmap Resetter utility was attempting to get storage for the global block.

**System action:** Processing stops with a return code of 12.

**User response:** Increase the region size and rerun the job.

**HPSB0104E** - OPEN FAILED FOR DDNAME: [HPSOUT | HPSOUT2]. TERMINATING

**Explanation:** The open failed for the data set that is shown in the message.

**System action:** Processing stops with a return code of 12.

**HPSB0105E** - [HPSOUT | HPSOUT2] DD STATEMENT MISSING. TERMINATING

**Explanation:** The required HPSOUT or HPSOUT2 DD is not specified.

**System action:** Processing stops with a return code of 12.

**User response:** Correct the error and rerun the job.

**HPSB0106E** - CREATE PC NUMBER PROCESS FAILED. ALL LIBRARIES IN STEPLIB CONCATENATION HAVE TO BE APF AUTHORIZED

**Explanation:** A load library of STEPLIB is not APF-authorized.

**System action:** Processing stops with a return code of 8.

**User response:** Correct the error and rerun the job.

**HPSB0107E** - GETMAIN FAILED FOR DATABASE DATA BLOCK

**Explanation:** A GETMAIN failed while attempting to get storage for the database data block.

**System action:** Processing stops with a return code of 8.

**User response:** Increase the region size and rerun the job.

**HPSB0108E** - LOAD FAILED FOR DFSBSCD0. TERMINATING

**Explanation:** After a LOAD macro was issued to load the module DFSBSCD0, the register 15 contained a nonzero return code.

**System action:** Processing stops with a return code of 12.

**User response:** Confirm DFSBSCD0 in IMS library is available, if not, correct the error and rerun the job.

**HPSB0109E** - ERROR RETURN FROM DBD MODULE

**Explanation:** The module which gets DBD information return an error internally.

**System action:** Processing stops with a return code of 8.
**User response:** The message which explains the cause of the error is displayed before this message. See that message and correct the error, then rerun the job.

**HPSB0110E - IMS RELEASE ERROR - IMS RELEASE NOT SUPPORTED. TERMINATING**

**Explanation:** The release level or maintenance level of IMS is lower than the level that is required for IMS HP Load.

**System action:** Processing stops with a return code of 12.

**User response:** Apply the required maintenance of IMS.

**HPSB0111E - ERROR RETURN FROM DYNAMIC ALLOCATION PROGRAM "$"**

**Explanation:** The database data set allocation program failed in allocation of the database data set which is displayed at "$", and returned an error.

**System action:** Processing is stopped with a return code of 8.

**User response:** Correct the error and rerun the job. If necessary, contact IBM Software Support.

**HPSB0112E - ERROR IN FREE SPACE CONTROL PROCESS PROGRAM**

**Explanation:** The bitmap I/O program returned an error.

**System action:** Processing is stopped with a return code of 8.

**User response:** The message which explains the cause of the error is displayed before this message. See the message and correct the error, then rerun the job. See the user response of the IMS message. Then, correct the error and rerun the job.

**HPSB0113E - ERROR IN HPIO INIT CALL**

**Explanation:** The INIT call of HPIO driver issued an error.

**System action:** Processing is stopped with a return code of 8.

**User response:** Contact IBM Software Support.

**HPSB0114E - ERROR IN HPIO TERMINATION CALL**

**Explanation:** The termination call of HPIO driver issued an error.

**System action:** Processing is stopped with a return code of 8.

**User response:** Contact IBM Software Support.

**HPSB0116E - ERROR RETURN FROM LIST.DB SCANNING PROGRAM**

**Explanation:** The LIST.DB program returned an error.

**System action:** Processing is stopped with a return code of 8.

**User response:** The message which explains the cause of the error is displayed before this message. See the message and correct the error, then rerun the job. See the user response of the IMS message. Then, correct the error and rerun the job.

**HPSB0118E - ERROR IN DBRC CONTROL PROCESS PROGRAM**

**Explanation:** HPSBDBRN which controls DBRC process program issued an error.

**System action:** Processing is stopped with a return code of 8.

**User response:** The message which explains the cause of the error is displayed before this message. See the message and correct the error, then rerun the job. See the user response of the IMS message. Then, correct the error and rerun the job.

**HPSB0119W - THE DUPLICATE PARTITION NAME (PARTNAME) SPECIFIED IN BMSETPART CONTROL STATEMENT**

**Explanation:** The duplicate partition name is specified in the BMSETPART control statement.

**System action:** IMS HP Load continues processing.

**User response:** Correct the partition name and rerun the job if necessary.

**HPSB0120E - MORE THAN 20 PARTITION NAMES ARE SPECIFIED IN THE BMSETPART CONTROL STATEMENT**

**Explanation:** Up to 20 partition names can be specified on the BMSETPART control statement; however, more than 20 names are specified.

**System action:** Processing stops with a return code of 8.

**User response:** Reduce the specification of partition names and rerun the job.

**HPSB0121E - YOU MUST SET DBRC=Y FOR HALDB**

**Explanation:** DBRC=N is specified though it is a HALDB.

**System action:** Processing stops with a return code of 8.
User response: Correct the error and rerun the job.

HPSB0122E - LOAD FAILED FOR HPSBRSTM.
TERMINATING
Explanation: Bitmap Resetter attempted to load HPSBRSTM but could not find it.
System action: Processing stops with a return code of 12.
User response: Correct the error and rerun the job. If the error persists, contact IBM Software Support.

HPSB0124E - CLOSE FAILED FOR DDNAME: ddname
Explanation: An error occurred while the non-VSAM data set that is allocated to the DD ddname was being closed. Use this optional control statement for a HALDB.
System action: Processing stops with a return code of 8.
User response: Correct the error and rerun the job.

HPSB0125E - THE BMSETPART CONTROL STATEMENT CANNOT BE SPECIFIED BY NON-HALDB
Explanation: When you restrict the partitions to be reset, use this optional control statement for a HALDB.
System action: Processing stops with a return code of 8.
User response: Correct the error and rerun the job.

HPSB0126E - THE VALUE FOR MORE THAN 10 DATA SET GROUPS IS SPECIFIED IN THE BMSETDSG CONTROL STATEMENT
Explanation: Up to 10 data set groups can be specified on the BMSETDSG control statement; however, more than 10 groups are specified.
System action: Processing stops with a return code of 8.
User response: Reduce the specification of partition names and rerun the job.

HPSB0127E - DO NOT USE A LARGE FORMAT D/S: dsname
Explanation: The dsname data set is a large format data set that is supported by z/OS Version 1 Release 7 and later. IMS HP Load cannot process large format data sets in z/OS Version 1 Release 6 or that of prior versions.
System action: Processing stops with a return code of 8.
User response: Check the level of z/OS.

HPSB0128E - GETDSAB FAILED FOR ddname
Explanation: The GETDSAB macro failed for DD name ddname.
System action: Processing stops with a return code of 8.
User response: Contact IBM Software Support.

HPSB0129E - SWAREQ FAILED FOR ddname
Explanation: The SWAREQ macro failed for DD name ddname.
System action: Processing stops with a return code of 8.
User response: Contact IBM Software Support.

HPSB0130E - OBTAIN FAILED FOR ddname
Explanation: The OBTAIN macro failed for DD name ddname.
System action: Processing stops with a return code of 8.
User response: Contact IBM Software Support.

HPSB0131E - ERROR RETURN FROM IGWASY (RC=return_code, RSN=reason_code)
Explanation: IGWASY returned an error return code. Here, RC shows the return code and RSN shows the reason code.
System action: Processing stops with a return code of 12.
User response: See DFSMSdfp Advanced Services and check the meaning of the return code and reason code.

HPSB0132W - IMSDALIB/STEPLIB MISSING, DYNAMIC ALLOCATION ERROR
Explanation: A database data set allocation error occurred because there is no specification of database data set and IMSDALIB or STEPLIB DD.
System action: Processing continues, return code 4.
User response: Correct the error and rerun the job.

HPSB0133W - NO MEMBER FOUND IN IMSDALIB/STEPLIB: member
Explanation: Specification of database data set and IMSDALIB or STEPLIB DD is not found.
System action: Processing continues, return code 4.
User response: Correct the error and rerun the job.

HPSB0134E - DYNAMIC ALLOCATION FAILED FOR DDNAME/DSNAME: ddname dsname
Explanation: An attempt at dynamic allocation for ddname and dsname failed.
System action: Processing stops with a return code of 8.
User response: Correct the error and rerun the job.

HPSB0135E - OPEN FAILED FOR DDNAME: ddname
Explanation: An open failure occurred because specification of the database data set is not found.
System action: Processing stops with a return code of 8.
User response: Correct the error and rerun the job.

HPSB0136E - MEMBER FOUND - ERROR IN NAME: member
Explanation: The DBD name that is specified for the parameter of program does not match with that of IMSDALIB DD.
System action: Processing stops with a return code of 8.
User response: Correct the error and rerun the job.

HPSB0137E - MEMBER FOUND - DDNAME MISSING: ddname
Explanation: The database data set name is not found in IMDALIB DD.
System action: Processing stops with a return code of 8.
User response: Correct the error and rerun the job.

HPSB0201E - OPEN FAILED FOR IMS DD
Explanation: The open failure of the IMS library occurred.
System action: Processing is stopped with a return code of 8.
User response: Contact IBM Software Support.

HPSB0202E - LOAD FAILED FOR DBD: dbdname
Explanation: A DBD load failure occurred.
System action: Processing is stopped with a return code of 8.
User response: Contact IBM Software Support.

HPSB0203E - DBD ORGANIZATION NOT HDAM/HIDAM OR PHDAM/PHIDAM
Explanation: The DBD information indicates that the database is not HDAM/HIDAM or PHDAM/PHIDAM.
System action: Processing is stopped with a return code of 8.
User response: Correct the error and rerun the job.

HPSB0204E - DBD NAME IN DBD NOT SAME AS MEMBER NAME
Explanation: The DBD name that is specified for the parameter of program is not the same as that of DBD definition.
System action: Processing is stopped with a return code of 8.
User response: Correct the error and rerun the job.

HPSB0301E - HPIO ERROR RETURN CODE - OPEN ERROR FOR DDNAME: ddname
Explanation: The OPEN call of HPIO driver issued an error.
System action: Processing is stopped with a return code of 8.
User response: Contact IBM Software Support.

HPSB0302E - HPIO ERROR RETURN CODE - CLOSE ERROR FOR DDNAME: ddname
Explanation: The CLOSE call of HPIO driver issued an error.
System action: Processing is stopped with a return code of 8.
User response: Contact IBM Software Support.

HPSB0303E - HPIO ERROR RETURN CODE - POINT ERROR FOR DDNAME: ddname
Explanation: The POINT call of HPIO driver issued an error.
System action: Processing is stopped with a return code of 8.
HPSB0304E - HPIO ERROR RETURN CODE
- READ ERROR FOR DDNAME: ddname

Explanation: The READ call of HPIO driver issued an error.

System action: Processing is stopped with a return code of 8.

User response: Contact IBM Software Support.

HPSB0305E - GETMAIN ERROR RETURN CODE FOR FREE SPACE ELEMENT

Explanation: The GETMAIN call for work area to update the bitmap block issued an error.

System action: Processing is stopped with a return code of 8.

User response: Contact IBM Software Support.

HPSB0306E - FREEMAIN ERROR RETURN CODE FOR FREE SPACE ELEMENT

Explanation: The FREEMAIN call for work area to update the bitmap block issued an error.

System action: Processing is stopped with a return code of 8.

User response: Contact IBM Software Support.

HPSB0307E - DBD ACCESS ERROR
- NOT OSAM/VSAM HDAM/HIDAM

Explanation: Unsupported database type is specified.

System action: Processing is stopped with a return code of 8.

User response: Contact IBM Software Support.

HPSB0311E - VSAM ERROR RETURN CODE
- GENRPL ERROR FOR DDNAME: ddname

Explanation: The GENRPL macro of VSAM issued an error.

System action: Processing is stopped with a return code of 8.

User response: Contact IBM Software Support.

HPSB0312E - VSAM ERROR RETURN CODE
- MODCB ERROR FOR DDNAME: ddname

Explanation: The MODCB macro of VSAM issued an error.

System action: Processing is stopped with a return code of 8.

User response: Correct the error and rerun the job.

HPSB0313E - VSAM ERROR RETURN CODE
- GET ERROR FOR DDNAME: ddname

Explanation: The GET macro of VSAM issued an error.

System action: Processing is stopped with a return code of 8.

User response: Contact IBM Software Support.

HPSB0314E - VSAM ERROR RETURN CODE
- PUT ERROR FOR DDNAME: ddname

Explanation: The PUT macro of VSAM issued an error.

System action: Processing is stopped with a return code of 8.

User response: Contact IBM Software Support.

HPSB0315E - BSAM ERROR RETURN CODE
- OPEN ERROR FOR DDNAME: ddname

Explanation: The OPEN macro of BSAM issued an error.

System action: Processing is stopped with a return code of 8.

User response: Contact IBM Software Support.

HPSB0316E - VSAM ERROR RETURN CODE
- OPEN ERROR FOR DDNAME: ddname

Explanation: The OPEN macro of VSAM issued an error.

System action: Processing is stopped with a return code of 8.

User response: Contact IBM Software Support.

HPSB0317E - CLOSE FAILED FOR DDNAME: ddname

Explanation: An error occurred while the non-VSAM data set that is allocated to the DD ddname was being closed.

System action: Processing stops with a return code of 8.

User response: Correct the error and rerun the job.

HPSB0501E - OPEN FAILED FOR LIST RECON DATA SET
- DDNAME: ddname

Explanation: A data set open error occurred for LIST.RECON.
HPSB0502E - NO INFORMATION FOUND IN RECON FOR DBD: dbdname
Explanation: No information is found in RECON for the DBD.
System action: Processing is stopped with a return code of 8.
User response: Contact IBM Software Support.

HPSB0604E - DBRC REQUEST FAILED FOR FUNCTION=INIT1
Explanation: The initialization call (FUNC=INIT1) of DBRC issued an error.
System action: Processing is stopped with a return code of 8.
User response: Contact IBM Software Support.

HPSB0505E - INCORRECT PARTNAME IN BMSETPART: partition
Explanation: An incorrect partition name is specified in the BMSETPART control statement.
System action: Processing stops with a return code of 8.
User response: Correct the error and rerun the job.

HPSB0605E - DBRC REQUEST FAILED FOR FUNCTION=SIGNON
Explanation: The DBRC SIGNON CALL (FUNC=SIGNON) issued an error.
System action: Processing is stopped with a return code of 8.
User response: Contact IBM Software Support.

HPSB0606E - GETMAIN FAILED FOR DBRC WORK AREA
Explanation: GETMAIN failed while attempting to get storage for the DBRC work area.
System action: Processing stops with a return code of 8.
User response: Determine how much more storage must be allocated for the job step. Increase the job-step region size, and rerun the job.

HPSB0607E - GETMAIN FAILED FOR DBRC CONTROL AREA
Explanation: GETMAIN failed while attempting to get storage for the DBRC control area.
System action: Processing stops with a return code of 8.
User response: Determine how much more storage must be allocated for the job step. Increase the job-step region size, and rerun the job.

HPSB0608E - GETMAIN FAILED FOR DBRC PARAMETER AREA
Explanation: GETMAIN failed while attempting to get storage for the DBRC parameter area.
System action: Processing stops with a return code of 8.
User response: Determine how much more storage must be allocated for the job step. Increase the job-step region size, and rerun the job.
HPSB0609E - GETMAIN FAILED FOR DBRC AUTHORIZATION LIST

Explanation: GETMAIN failed while attempting to get storage for the DBRC authorization list.

System action: Processing stops with a return code of 8.

User response: Determine how much more storage must be allocated for the job step. Increase the job-step region size, and rerun the job.

HPSB0610E - DBRC REQUEST FAILED FOR FUNCTION=SIGNOFF

Explanation: The DBRC SIGNOFF call (FUNC=SIGNOFF) of DBRC issued an error.

System action: Processing is stopped with a return code of 8.

User response: Contact IBM Software Support.

HPSB0611E - DBRC REQUEST FAILED FOR FUNCTION=TERM

Explanation: The DBRC termination call (FUNC=TERM) of DBRC issued an error.

System action: Processing is stopped with a return code of 8.

User response: Contact IBM Software Support.

HPSB0631E - DBRC REQUEST FAILED, ERROR RETURN CODE FROM DFSDBAU0

Explanation: The DBRC authorization call (DFSDBAU0) of DBRC issued an error.

System action: Processing is stopped with a return code of 8.

User response: Contact IBM Software Support.

HPSB0641E - DBRC REQUEST FAILED FOR FUNCTION=UNAUTHORIZE

Explanation: The DBRC authorization call (FUNC=UNAUTH) of DBRC issued an error.

System action: Processing is stopped with a return code of 8.

User response: Contact IBM Software Support.

HPSB3901E OPEN FAILED FOR DDNAME: ddname

Explanation: After an OPEN macro is issued to open the non-VSAM data set that is associated with ddname, DCBOFLGS becomes nonzero.

System action: Processing stops.

User response: Ensure that the DD statement associated with ddname indicates the correct data set. Correct the error and rerun the job.

HPSB3902E CONTROL STATEMENT SYNTAX ERROR

Explanation: An incorrect control statement is detected by the control statement syntax checking.

System action: Processing stops.

User response: Check the control statement. Correct the error and rerun the job.

HPSB3903E INCORRECT PARAMETER SPECIFIED: parm_string

Explanation: An incorrect parameter is detected. The incorrect parameter string is listed as parm_string.

System action: Processing stops.

User response: Check the parameter specification that is listed as parm_string. Correct the parameter error and rerun the job.

HPSB3904E NUMBER OF parm_name PARAMETERS EXCEEDED THE LIMIT: nnn

Explanation: An incorrect control statement is detected by the control statement syntax checking.

System action: Processing stops.

User response: Check the control statement. Correct the error and rerun the job.

HPSB3905E NUMBER OF OPERANDS FOR parm_name PARAMETER EXCEEDED THE LIMIT: nnn

Explanation: An incorrect control statement is detected by the control statement syntax checking.

System action: Processing stops.

User response: Check the control statement. Correct the error and rerun the job.

HPSB3906E LENGTH ERROR IN n-th OPERAND OF PARAMETER: parm_name

Explanation: An incorrect control statement is detected by the control statement syntax checking.

System action: Processing stops.

User response: Check the control statement. Correct the error and rerun the job.
HPSB3907E  ATTRIBUTE ERROR IN n-th OPERAND OF PARAMETER: parm_name

Explanation: An incorrect control statement is detected by the control statement syntax checking.

System action: Processing stops.

User response: Check the control statement. Correct the error and rerun the job.

HPSB3908E  NUMBER OF VALUES FOR n-th OPERAND EXCEEDED THE LIMIT: nnn

Explanation: An incorrect control statement is detected by the control statement syntax checking.

System action: Processing stops.

User response: Check the control statement. Correct the error and rerun the job.

HPSB3909E  n-th OPERAND IS REQUIRED FOR PARAMETER: parm_name

Explanation: An incorrect control statement is detected by the control statement syntax checking.

System action: Processing stops.

User response: Check the control statement. Correct the error and rerun the job.

HPSB3910I  BLANK LINE IN HPSIN

Explanation: A blank line is read in the HPSIN data set.

System action: Processing continues.

User response: None. This message is informational.

HPSB3999E  GETMAIN FAILED FOR WORK AREA

Explanation: Program GETMAIN failed while attempting to get storage for the work area.

System action: IMS HP Load issues a user 3999 abend.

User response: Determine how much more storage must be allocated for the job step. Increase the job-step region size, and rerun the job.
Chapter 20. Gathering diagnostic information

Before you report a problem with IMS HP Load to IBM Software Support, you need to gather the appropriate diagnostic information.

Procedure

Provide the following information for all IMS HP Load problems:

• A clear description of the problem and the steps that are required to re-create the problem
• The version of IMS that you are using and the version of the operating system that you are using
• A complete log of the job
• A Load Module/Macro APAR Status report

For information about creating a Load Module/Macro APAR Status report, see Chapter 21, “Diagnostics Aid,” on page 321.
Chapter 21. Diagnostics Aid

If you have a problem that you think is not a user error, run the Diagnostics Aid (FABFDIAG), obtain the Load Module/Macro APAR Status report, attach it to the other diagnostic documents (such as job dump list or I/O of the utility), and report the error to IBM.

The Diagnostics Aid generates Load Module/Macro APAR Status reports. These reports show the latest APAR fixes applied to each module and macro.

The Diagnostics Aid is not applicable for any other versions or releases.

Topics:
- “Running the Diagnostics Aid with JCL” on page 322
- “Load Module/Macro APAR Status report” on page 323
- “Messages and codes” on page 325
Running the Diagnostics Aid with JCL

To run the Diagnostics Aid program (FABFDIAG), supply an EXEC statement and DD statements that define the input and the output data sets.

Procedure

1. Specify the EXEC statement. It must be in the following form:
   
   //stepname EXEC PGM=FABFDIAG

2. Specify the DD statements.

<table>
<thead>
<tr>
<th>DD statement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>STEPLIB DD</td>
<td>This statement defines the library that contains the load modules (usually HPS.SHPSLMD0).</td>
</tr>
<tr>
<td>SHPSLMD DD</td>
<td>This statement defines the library that contains the load modules (usually HPS.SHPSLMD0) for which you have a problem.</td>
</tr>
<tr>
<td></td>
<td>The Load Module APAR Status report is not generated if this DD statement is not provided or if DD DUMMY is specified.</td>
</tr>
<tr>
<td></td>
<td>It is always recommended that you specify this DD statement.</td>
</tr>
<tr>
<td>SHPSMAC DD</td>
<td>This statement defines the library that contains the provided macros (usually HPS.SHPSMAC0) for which you have a problem.</td>
</tr>
<tr>
<td></td>
<td>The Macro APAR Status report is not generated if this DD statement is not provided or if DD DUMMY is specified.</td>
</tr>
<tr>
<td>SYSPRINT DD</td>
<td>This output data set contains the Load Module/Macro APAR Status report. The data set contains 133-byte, fixed-length records. It can reside on a tape, direct-access device, or printer; or it can be routed through the output stream. If BLKSIZE is coded in the DD statement, it must be a multiple of 133. However, it is recommended that you use:</td>
</tr>
<tr>
<td></td>
<td>//SYSPRINT DD SYSOUT=A</td>
</tr>
</tbody>
</table>

3. Run the job.
Load Module/Macro APAR Status report

The Diagnostics Aid generates two reports for maintenance by IBM.

The generated reports are:
• Load Module APAR Status report
• Macro APAR Status report

Load Module APAR Status report

The Load Module APAR Status report contains information about the modules and their applied APARs.

This report contains the following information:

MODULE LIBRARY
This field includes the data set names that are specified in the SHPSLMD DD statement. If more than 30 data sets are concatenated, only the first 30 data sets are listed.

MODULE NAME
This field shows the name of the load module member or the alias.

ALIAS-OF
This field shows the name of the original member of the alias. If the module name is not an alias, this field is left blank.

CSECT NAME
This field shows the name of the included CSECT in the module. The CSECT names are reported in the included order in the module.

APAR NUMBER
This field shows the latest APAR number that is applied to the module represented by the CSECT name. If no APAR is applied, NONE is shown.

APAR FIX-DATE
This field shows the date when the modification was prepared for the module that is represented by the CSECT name. If no APAR is applied, N/A is shown.

Notes:
1. If the CSECT name does not start with FAB, HPS, or the program structure of the CSECT does not conform to the IMS HP Load module standard to identify the APAR number and the APAR fixed date, the fields APAR NUMBER and APAR FIX-DATE are filled with asterisks (*).
2. If the load module is a member of the PDSE library, the following statement is shown on the report line and the job completes with a return code of 4.
   ** IT CAN NOT BE ANALYZED DUE TO PDSE LIBRARY MEMBER **
3. If the load macro fails for a utility member, the following statement is shown on the report line and the job completes with a return code of 8.
   ** IT CAN NOT BE ANALYZED DUE TO LOAD FAILED MEMBER **
**Macro APAR Status report**

The Macro APAR Status report contains information about macros and their applied APARs.

This report contains the following information:

**MACRO LIBRARY**
This field includes the data set names that are specified in the SHPSMAC DD statement. If more than 30 data sets are concatenated, only the first 30 data sets are listed.

**MACRO NAME**
This field shows the name of the macro member or the alias.

**ALIAS-OF**
This field shows the name of the original member of the alias. If the macro name is not an alias, this field is left blank.

**APAR NUMBER**
This field shows the latest APAR number that is applied to the macro. If no APAR is applied, NONE is shown.

**APAR FIX-DATE**
This field shows the date when the modification was prepared for the macro. If no APAR is applied, N/A is shown.

**Note:** If the macro source statement structure does not conform to the IMS HP Load macro standard to identify the APAR number and the APAR fixed date, the fields APAR NUMBER and APAR FIX-DATE are filled with asterisks (*).
Messages and codes

The following topics explain the messages and codes that are issued by the Diagnostics Aid (FABFDIAG).

Return codes

FABFDIAG contains the following return codes:

- **0**: Successful completion of the program.
- **4**: Warning messages were issued, but the requested operation was completed.
- **8**: Error messages were issued, but the request operation was completed.

Abend codes

All 36xx abend codes are accompanied by an FABU36xx message. Locate the associated message for problem determination.

Messages

Use the information in these messages to help you diagnose and solve FABFDIAG problems.

<table>
<thead>
<tr>
<th>Message Code</th>
<th>Message Description</th>
<th>Explanation</th>
<th>System action</th>
<th>User response</th>
</tr>
</thead>
<tbody>
<tr>
<td>FABU1001I</td>
<td>DIAG ENDED NORMALLY</td>
<td>This message is issued when Diagnostic Aid ends successfully.</td>
<td>Diagnostic Aid completes the job successfully with a return code of 0.</td>
<td>None. This message is informational.</td>
</tr>
<tr>
<td>FABU1002W</td>
<td>DIAG ENDED WITH WARNINGS</td>
<td>This message is generated when trivial error conditions are encountered by Diagnostic Aid.</td>
<td>Diagnostic Aid ends with a return code of 4.</td>
<td>See other messages that are generated by Diagnostic Aid to determine the nature and the cause of the detected errors. Correct the problem and rerun the job.</td>
</tr>
<tr>
<td>FABU1003E</td>
<td>DIAG ENDED WITH ERRORS</td>
<td>This message is generated when severe error conditions are encountered by Diagnostic Aid.</td>
<td>Diagnostic Aid ends with a return code of 8.</td>
<td>See other messages that are generated by Diagnostic Aid to determine the nature and the cause of the detected errors. Correct the problem and rerun the job.</td>
</tr>
<tr>
<td>FABU1005W</td>
<td>[SHPSLMD</td>
<td>SHPSMAC] DD STATEMENT NOT FOUND</td>
<td>Diagnostic Aid could not find the SHPSLMD/SHPSMAC DD statement.</td>
<td>Diagnostic Aid sets an end-of-job return code of 4 and continues processing. Diagnostic Aid does not generate a report for the load module or the macro.</td>
</tr>
<tr>
<td>FABU1007W</td>
<td>DUMMY SPECIFIED FOR [SHPSLMD</td>
<td>SHPSMAC] DD STATEMENT</td>
<td>DUMMY is specified for the SHPSLMD/SHPSMAC DD statement.</td>
<td>Diagnostic Aid sets an end-of-job return code of 4 and continues processing. Diagnostic Aid does not generate a report for the load module or the macro.</td>
</tr>
<tr>
<td>Error Code</td>
<td>Description</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FABU1008W</td>
<td>**NO [MODULE</td>
<td>MACRO] MEMBERS FOUND IN DDNAME [SHPSLMD</td>
<td>SHPSMAC]**</td>
<td></td>
</tr>
<tr>
<td><strong>Explanation:</strong></td>
<td>Diagnostic Aid could not find any utility modules or macros members from the DD ddname data set.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>System action:</strong></td>
<td>Diagnostic Aid sets an end-of-job return code of 4 and continues processing.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>User response:</strong></td>
<td>Ensure that the libraries have correct utility module or macro libraries. Correct the error and rerun the job.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FABU2001E</td>
<td><strong>LOAD FAILED FOR DDNAME ddname MODULE member</strong></td>
</tr>
<tr>
<td><strong>Explanation:</strong></td>
<td>Diagnostic Aid could not load the indicated member from ddname.</td>
</tr>
<tr>
<td><strong>System action:</strong></td>
<td>Diagnostic Aid sets an end-of-job return code of 8 and continues processing.</td>
</tr>
<tr>
<td><strong>User response:</strong></td>
<td>Ensure that the indicated member exists in the data set that is specified for the indicated ddname. Correct the error and rerun the job.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FABU3600E</td>
<td><strong>OPEN FAILED FOR DDNAME ddname</strong></td>
</tr>
<tr>
<td><strong>Explanation:</strong></td>
<td>The named DCB could not be opened.</td>
</tr>
<tr>
<td><strong>System action:</strong></td>
<td>Diagnostic Aid ends with an abend code of U3600.</td>
</tr>
<tr>
<td><strong>User response:</strong></td>
<td>Ensure that a ddname DD statement exists, and that it specifies the correct DD parameter. Correct any errors, and rerun the job.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FABU3601E</td>
<td><strong>GET FAILED FOR DDNAME ddname</strong></td>
</tr>
<tr>
<td><strong>Explanation:</strong></td>
<td>The GET failed for a directory from the DD ddname data set.</td>
</tr>
<tr>
<td><strong>System action:</strong></td>
<td>Diagnostic Aid ends with an abend code of U3601.</td>
</tr>
<tr>
<td><strong>User response:</strong></td>
<td>See the MVS system message and its programmer response. Correct the error and rerun Diagnostic Aid. If the error persists, contact IBM Software Support.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FABU3602E</td>
<td><strong>READ FAILED FOR DDNAME ddname MEMBER member</strong></td>
</tr>
<tr>
<td><strong>Explanation:</strong></td>
<td>The READ failed for a member from the DD ddname data set.</td>
</tr>
<tr>
<td><strong>System action:</strong></td>
<td>Diagnostic Aid ends with an abend code of U3602.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FABU3603E</td>
<td><strong>BLDL FAILED FOR DDNAME ddname MEMBER member</strong></td>
</tr>
<tr>
<td><strong>Explanation:</strong></td>
<td>The member was not found when the BLDL macro searched the PDS directory for the ddname.</td>
</tr>
<tr>
<td><strong>System action:</strong></td>
<td>Diagnostic Aid ends with an abend code of U3603.</td>
</tr>
<tr>
<td><strong>User response:</strong></td>
<td>Ensure that the member indicated exists in the data set that is specified for the indicated ddname. Correct the error and rerun the job. If the error persists, contact IBM Software Support.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FABU3604E</td>
<td><strong>LOAD FAILED FOR DDNAME ddname MODULE member</strong></td>
</tr>
<tr>
<td><strong>Explanation:</strong></td>
<td>Diagnostic Aid could not load the member name from the ddname.</td>
</tr>
<tr>
<td><strong>System action:</strong></td>
<td>Diagnostic Aid ends with an abend code of U3604.</td>
</tr>
<tr>
<td><strong>User response:</strong></td>
<td>See the MVS system message and its programmer response. Correct the error and rerun Diagnostic Aid. If the error persists, contact IBM Software Support.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FABU3605E</td>
<td><strong>DELETE FAILED FOR MODULE member</strong></td>
</tr>
<tr>
<td><strong>Explanation:</strong></td>
<td>Diagnostic Aid could not delete a member name.</td>
</tr>
<tr>
<td><strong>System action:</strong></td>
<td>Diagnostic Aid ends with an abend code of U3605.</td>
</tr>
<tr>
<td><strong>User response:</strong></td>
<td>Contact IBM Software Support.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FABU3606E</td>
<td><strong>PUT FAILED FOR SYSPRINT</strong></td>
</tr>
<tr>
<td><strong>Explanation:</strong></td>
<td>Diagnostic Aid could not put report data in SYSPRINT.</td>
</tr>
<tr>
<td><strong>System action:</strong></td>
<td>Diagnostic Aid ends with an abend code of U3606.</td>
</tr>
<tr>
<td><strong>User response:</strong></td>
<td>See the MVS system message and its programmer response. Correct the error and rerun Diagnostic Aid. If the error persists, contact IBM Software Support.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FABU3607E</td>
<td><strong>OPEN FAILED FOR SYSPRINT</strong></td>
</tr>
<tr>
<td><strong>Explanation:</strong></td>
<td>SYSPRINT DCB could not be opened.</td>
</tr>
<tr>
<td><strong>System action:</strong></td>
<td>Diagnostic Aid ends with an abend code of U3607.</td>
</tr>
</tbody>
</table>
| **User response:** | Ensure that a ddname SYSPRINT DD statement exists, and that it specifies the correct DD
FABU3608E  FIND FAILED FOR DDNAME \texttt{ddname} MEMBER \texttt{member}

\textbf{Explanation:} The FIND failed for a member from DDNAME \texttt{ddname} data set.

\textbf{System action:} Diagnostic Aid ends with an abend code of U3608.

\textbf{User response:} Ensure that the member indicated exists in the data set that is specified for the indicated ddname. Correct the error and rerun the job. If the error persists, contact IBM Software Support.

FABU3609E  DEVTYPE FAILED FOR DDNAME \texttt{ddname}

\textbf{Explanation:} The DEVTYPE failed for DDNAME \texttt{ddname} data set.

\textbf{System action:} Diagnostic Aid ends with an abend code of U3609.

\textbf{User response:} Contact IBM Software Support.

FABU3610E  RDJFCB FAILED FOR DDNAME \texttt{ddname}

\textbf{Explanation:} The READJFCB failed for DDNAME \texttt{ddname} data set.

\textbf{System action:} Diagnostic Aid ends with an abend code of U3610.

\textbf{User response:} Contact IBM Software Support.

FABU3611E  GETMAIN FAILED. INSUFFICIENT STORAGE TO RUN THE JOB

\textbf{Explanation:} Workspace for Diagnostic Aid could not be obtained.

\textbf{System action:} Diagnostic Aid ends with an abend code of U3611.

\textbf{User response:} Increase the region size and rerun the job.

FABU3612E  TOO MANY [MODULE | MACRO] MEMBERS DETECTED IN DDNAME [SFABMOD | SHPSMAC]

\textbf{Explanation:} There are too many utility members in the SFABMOD/SHPSMAC DD data set.

\textbf{System action:} Diagnostic Aid ends with an abend code of U3612.

\textbf{User response:} Specify the correct data set for the indicated DD statement and rerun the job.
Part 8. Appendixes
Notices

This information was developed for products and services offered in the U.S.A.

This material may be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

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 give 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 (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.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, 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.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

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 Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US.

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 information 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.

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.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.
Programming interface information

This publication primarily documents information that is NOT intended to be used as Programming Interfaces of IMS High Performance Load.

This publication also documents intended general-use programming interface and associated guidance information, product-sensitive programming interface and associated guidance information, and diagnosis, modification, or tuning information for IMS High Performance Load.

Product-sensitive programming interfaces are provided to allow the customer installation to perform tasks such as tailoring, monitoring, modification, or diagnosis of the IBM product. Use of such interfaces creates dependencies on the detailed design or implementation of the IBM product. Product-sensitive interfaces should be used only for these specialized purposes. Because of their dependencies on detailed design and implementation, it is to be expected that programs written to such interfaces might need to be changed in order to run with new product releases or versions, or as a result of service. This information is identified where it occurs, either by an introductory statement to a topic or by the following marking:

PSPI

Product-sensitive programming interface and associated guidance information...

PSPI

Diagnosis, modification, or tuning information is provided to help the customer diagnose, modify, or tune IMS High Performance Load. This information is identified where it occurs, either by an introductory statement to a topic or by the following marking:

DMTI

Diagnosis, modification, or tuning information...

DMTI

Attention: Do not use this diagnosis, modification, or tuning information as a programming interface.
Trademarks

IBM, the IBM logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at [http://www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

Java™ and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Other company, product, and service names may be trademarks or service marks of others.
Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability

These terms and conditions are in addition to any terms of use for the IBM website.

Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.
Privacy policy considerations

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering’s use of cookies is set forth below.

This Software Offering does not use cookies or other technologies to collect personally identifiable information.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

### Index

#### A
- abend codes 253
- accessibility 21
- Actual Root per Block report 222
- Assigned Roots per Block report 199, 219
- Assigned Roots per Rap report 219
- Assigned Roots per Rap report 198

#### B
- Bitmap Reset Process Summary report 227
- Bitmap Reseter 227
- BMSETDSG control statement 49
- Bitmap Reseter 234
- BMSETPART control statement
- Bitmap Reseter 235
- BUFFERNO control statement 50
- BUFNO control statement 50

#### C
- cascading 156, 179, 197, 215
- CHKEMPTYUL control statement 50
- installation default option 244
- CHKTRAIL control statement 51
- installation default option 244
- CHKUPDT control statement 52
- installation default option 244
- COBOL 24
  - IMS HP Load 110
- Load API 137
- PSSR 202
- Comments on DL/I HDAM DB Load Procedure report 194, 197
- COMP control statement 53
- compatibility
- IMS HP Load 15
- COMPAUTH control statement 52
- installation default option 244
- COMPRESS control statement 53
- COMPRESSED control statement 202
- FABSPSSI PSSR 182
- FABSPSSR PSSR 177
- compression factor 97
- Control Card Format report 194, 195
- Control Card report 194, 196
- control statements
  - Bitmap Reseter 234
  - IMS HP Load 45
  - Load API 146
  - PSSR 175
- cookie policy 331, 336
- corrupted database 117
- CTL data set 171, 178

#### D
- data flow
  - IMS HP Load 32
  - Load API 142
  - PSSR 163
- Data Set Statistics report 89
- data sets (Bitmap Resetter)
  - database 232
  - HPSIN 232
  - HPSOUT 232
  - HPSOUT2 232
  - IMS 232
  - IMSDALIB 232
  - RECON 232
  - STEPLIB 231
  - SYSUDUMP 233
- data sets (IMS HP Load)
  - database 38
  - DDEPPDS 38
  - DFSRESLB 37
  - DFSUNITPT 37
  - DFSURCDS 37
  - DFSURWF1 39
  - DFSVSAMP 38
  - FRRIN 38, 45
  - HPSRAMSO 42
  - HPSRISMG 42
  - IMS 37
  - PROCLIB 37
  - RECON 38
  - SEQERROR 42
  - SNAP 42, 103
  - SORTWK2s 42
  - STEPLIB 36
  - SYSPRINT 42
  - SYSUDUMP 42
- data sets (Load API)
  - DFSRESLB 144
  - DFSVSAMP 144
  - FRRIN 144
  - IMS 143
  - SORTWK2s 144
  - STEPLIB 143
  - SYSOUT 144
  - SYSPRINT 144
- data sets (PSSR)
  - CTL 171, 178, 183
  - DFSRESLB 174
  - HDHEADER 173
  - IMS 172
  - IMS2 172, 202
  - output partition data set 173, 194
  - PR10 200
  - PRS 173, 194
  - PR9 173, 198
  - PR9X 173
  - PSSR 171, 177, 182, 202
  - RECON 174
  - SORTCNTL 171, 181
  - SORTOUT 172, 191, 192
  - SORTOUT2 172, 191, 192
- data sets (PSSR) (continued)
  - SORTOUT3 172, 193
  - SORTOUT4 173, 193
  - SORTOUTD 173, 193
  - SORTWK2s 173
  - STEPLIB 171
  - SYSABEND 173
  - SYSDDUMP 173
  - SYSIN 171, 175
  - SYSOUT 173, 194
  - SYSPRINT 174, 194
- database 38
- database DD statement
  - Bitmap Reseter 232
  - IMS HP Load 38
- Database Image Copy utility 166
- Database Prefix Resolution utility 166
- Database Prefix Update utility 166
- Database Reorganization utility 166
- Database Scan utility 166
- DATXEXIT
  - PSSR 171
- DATXEXIT control statement 54
  - FABSPSSI PSSR 182
- DATXEXIT= IMS HP Load 54
- DB statement 183
- DBD 166, 212, 213, 215, 217, 219
  - PSSR 163
  - DBD Parameters and Overrides report 186, 194, 196
- DBD Snap Dump report 103
- DBDCEN 166
- DBDSNAP control statement 55
- DBIOBUF control statement 55
- DBRC command authorization 27
- DBRCGRP= 171
- DBRLBUF control statement 56
- DDEPPDS data set
  - IMS HP Load 38
- DFSORT 166, 171, 175, 181, 194
  - PSSR 163
- DFSRESLB data set
  - IMS HP Load 37
  - Load API 144
  - PSSR 174
- DFSRRCC00 36, 58
  - PSSR 163
- DFSUNITPT data set 37
- DFSUPNTO (IMS HALDB Partition Initialization utility) 10, 110
- DFSURCDS data set 37
- DFSURGL0 (IMS HD Reorganization Reload utility) 9, 166, 281
- DFSURGS0 (IMS Database Scan utility) 39, 166
- DFSURGL0 (IMS HD Reorganization Unload utility) 9, 15, 156, 166
- DFSURPR0 (IMS Database Reorganization utility) 37, 110, 166
- DFSURWF1 data set 39

---

337
HD Tuning Aid utility 219, 222
IMS HP Load 9
HALDB Reorganization 108
installation default option 244
HDAM 9
HALDB partition data set initialization utility 10
HDAM 9
PSSR 159, 162
HALDB Process Summary report
HDAM 9
PSSR 162
HALDB partition data set initialization utility 10
HALDB Reorganization 108
installation default option 244
HIDAM 9
IMS HP Load 9
HALDB Reorganization 108
installation default option 244
High Availability Large Database (HALDB) 9
IMS HP Load 9
HALDB Reorganization 108
installation default option 244
High Speed Sequential Retrieval 156, 166, 171, 212, 217, 219, 222
HISAM 9
IMS HP Load 9
HALDB Reorganization 108
installation default option 244
HISAM 9
H alte r Key
HALDB Reorganization 108
installation default option 244
HPSB messages 310
HALDB Reorganization 108
installation default option 244
HPSBIRSET 231
HALDB Reorganization 108
installation default option 244
HPSIN Control Statements report 236
HALDB Reorganization 108
installation default option 244
HPSIN data set 232
HALDB Reorganization 108
installation default option 244
HPSOUT data set 232
HALDB Reorganization 108
installation default option 244
HPSOUT2 data set 232
HALDB Reorganization 108
installation default option 244
HPSR messages 259
HALDB Reorganization 108
installation default option 244
HPSRAMSO data set
HALDB Reorganization 108
installation default option 244
IMS HP Load 42
HALDB Reorganization 108
installation default option 244
HPSRISMG data set 42
HALDB Reorganization 108
installation default option 244
HPSROPT macro statements 244
HALDB Reorganization 108
installation default option 244
HPSRRC00 58
HALDB Reorganization 108
installation default option 244
HPSRVS10 exit 28
HALDB Reorganization 108
installation default option 244
HPSKTRAC data set 144
HALDB Reorganization 108
installation default option 244
IMS HALDB Partition Initialization utility 110
IMS HD Reorganization 108
installation default option 244
IMS HP Load 9
IMS HD Reorganization 108
installation default option 244
IMS HP Load 56
IMS HP Load 9
IMS HALDB Partition Initialization utility 110
IMS HD Reorganization 108
installation default option 244
IMS HISAM Reorganization Utility 166
IMS HISAM Reorganization Utility 166
IMS HP Load utility 9
IMS Tools Knowledge Base configuration 25
IMS/ESA Year 2000 Exit Tool
IMS HP Load 24
PSSR 162
IMS2 data set 172, 202
IMS/ESA Year 2000 Exit Tool
IMS HP Load 45
Load API 146
PSSR 175, 181
INPUT control statement 60
installation default option 244
introduction 3
Batch Resetter 237
IMS HP Load 9
Load API 137
PSSR 155
ITKBDLOAD control statement 61
installation default option 244
ITKBSRVR control statement 61
installation default option 244
ITKBSRVR control statement 61
installation default option 244
JCL requirements
Batch Resetter 231
IMS HP Load 35
Load API 143
PSSR 170
job control language (JCL)
DFSORT
PSSR 170
procedure 166
PSSR 170
job steps
PSSR 166
K
keyboard shortcuts 21
L
Language Environment 24
IMS HP Load 60, 119
Load API 137
PSSR 202
partition high key
OVERFLOW data set
output partition data set
operating instructions
old and new content of bitmap
load api
load api prerequisite
logical relationships
LPART control statement

M
messages
methods for accessing
overview
messages and codes
MODS control statement
FABSPSSR SYSIN

N
notices

O
OADSPR control statement
installation default option
Old and New Content of Bitmap report
operating instructions
PSSR
output
Bitmap Resetter
IMS HP Load
Load API
PSSR
output partition data set
OVERFLOW data set

P
PAD control statement
installation default option
Parameter and DBD/Data Sets report
PART statement
PARTINIT control statement
installation default option
partition
IMS HP Load
PSSR
partition high key
IMS HP Load
PSSR
partition initialization
IMS HP Load
PSSR
partition selection exit
IMS HP Load
PSSR
partition selection string
IMS HP Load
PSSR
PARTSTAT control statement
PDHAM
IMS HP Load
PSSR
PDHAM
IMS HP Load
PSSR
PIPEWAIT control statement
PL/1
IMS HP Load
Load API
PSSR
Pointe
<table>
<thead>
<tr>
<th>Unloaded data set format (PSSR)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS format 171, 178, 182</td>
</tr>
<tr>
<td>HD format 171</td>
</tr>
<tr>
<td>SH format 171</td>
</tr>
<tr>
<td>user exit 10</td>
</tr>
<tr>
<td>IMS HP Load 10, 119</td>
</tr>
<tr>
<td>PSSR 182, 202</td>
</tr>
<tr>
<td>USEREXIT control statement</td>
</tr>
<tr>
<td>FABPSSI PSSRIN 182</td>
</tr>
<tr>
<td>IMS HP Load 73</td>
</tr>
<tr>
<td>USERHDR control statement 73</td>
</tr>
<tr>
<td>USERHDRX control statement 75</td>
</tr>
</tbody>
</table>

### W

<table>
<thead>
<tr>
<th>WF1DSN control statement 75</th>
</tr>
</thead>
<tbody>
<tr>
<td>WFREC control statement 76</td>
</tr>
<tr>
<td>installation default option 244</td>
</tr>
<tr>
<td>WFI SPACE control statement 77</td>
</tr>
<tr>
<td>WFI VOL control statement 77</td>
</tr>
<tr>
<td>WFP data sets 133</td>
</tr>
<tr>
<td>WFP Process Summary report 102</td>
</tr>
<tr>
<td>WFP HLQ control statement 78</td>
</tr>
<tr>
<td>WFPS SPACE control statement 78</td>
</tr>
<tr>
<td>WFP UNIT control statement 79</td>
</tr>
<tr>
<td>WFP VOL control statement 79</td>
</tr>
<tr>
<td>WFP VOL CNT control statement 80</td>
</tr>
</tbody>
</table>