CICS Transaction Server for z/OS
Version 4 Release 2

Supplied Transactions

SC34-7184-00

CICS Transaction Server for z/OS
Version 4 Release 2

Supplied Transactions

SC34-7184-00

Note
Before using this information and the product it supports, read the information in "Notices" on page 643.

This edition applies to Version 4 Release 2 of CICS Transaction Server for z/OS (product number 5655-S97) and to all subsequent releases and modifications until otherwise indicated in new editions.

© Copyright IBM Corporation 1982, 2011.
US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Preface vii
What this book is about vii
Who should read this book vii
What you need to know to understand this book vii
How to use this book vii

Changes in CICS Transaction Server for z/OS, Version 4 Release 2 ix

Chapter 1. Using CICS supplied transactions 1
How to start or stop a transaction 1
Syntax notation and conventions used 2
Minimum abbreviation of keywords 3
Uppercase input to transactions 3
Terminal operation 4
$ (the dollar symbol) 4
Operator security 4
Terminal operator. 4
Supervisory terminal operator 5
Master terminal operator 6
MRO and ISC support 7
BMS partitions. 7
Using the system console 9
Rules for console entry 9
MODIFY command 9
REPLY command 10
Example of a conversation using CONVERSE . . 11
Example of a conversation using WRITE/READ 11
Using TSO consoles. 11
Using the KILL option for purging transactions . . 12
Using CEMT to terminate a task 12
Using the console to terminate a task. 13
Consequences of using KILL 13

Chapter 2. The application debugging profile manager 15
Using the application debugging profile manager Web interface. 15
Configuring access to the application debugging profile manager Web interface 15
Creating a debugging profile with the Web interface 21
Activating debugging profiles with the Web interface 22
Inactivating debugging profiles with the Web interface 23
Viewing a debugging profile with the Web interface 23
Changing a debugging profile with the Web interface 24
Copying debugging profiles with the Web interface 24
Deleting debugging profiles with the Web interface 25
Setting the display device 34
Using the application debugging profile manager 3270 interface. 37
Creating a debugging profile with the 3270 interface 40
Activating debugging profiles with the 3270 interface 41
Inactivating debugging profiles with the 3270 interface 42
Viewing a debugging profile with the 3270 interface 42
Changing a debugging profile with the 3270 interface 42
Copying debugging profiles with the 3270 interface 43
Deleting debugging profiles with the 3270 interface 44
Combining actions on the List debugging profiles screen 44
Setting the display device 55

Chapter 3. CBAM - BTS browser. 59

Chapter 4. CCRL - certificate revocation lists transaction 61

Chapter 5. CDBC - database control menu 63
Connecting CICS to DBCTL 64
Disconnecting CICS from DBCTL 65
Specifying a DBCTL system identifier (DBCTLID) 65

Chapter 6. CDBI - database control inquiry 67

Chapter 7. CDBM - database control interface 69
Using CDBM Syntax 69
Using DFHDBFK - The CDBM GROUP command file 72
Record layout in the CDBM GROUP command file 72
The MAINTENANCE panel for DFHDBFK. . . 73

Chapter 8. CEBR - temporary storage browse. 75

Chapter 9. CEBT - master terminal (alternate CICS) 77
Minimum abbreviation of keywords 77
CEBT INQUIRE and SET commands 77

© Copyright IBM Corp. 1982, 2011

CEBT INQUIRE AUXTRACE 77
CEBT INQUIRE DUMP 79
CEBT INQUIRE GTFTRACE. 79
CEBT INQUIRE INTTRACE 80
CEBT INQUIRE SURVEILLANCE 81
CEBT INQUIRE TAKEOVER 81
CEBT SET AUXTRACE 81
CEBT SET DUMP 82
CEBT SET GTFTRACE 83
CEBT SET INTTRACE. 84
CEBT SET SURVEILLANCE. 85
CEBT SET TAKEOVER 85
CEBT PERFORM commands 86
CEBT PERFORM SHUTDOWN 86
CEBT PERFORM SNAP 86
CEBT PERFORM TAKEOVER 87

Chapter 10. CECI - command-level interpreter 89
Command-level syntax-checking transaction (CECS) 89

Chapter 11. CEDA - resource definition online 91
Command syntax 91
CSD compatibility panel 95
Security checking 96
CEDB and CEDC 97

Chapter 12. CEDF and CEDX - the execution diagnostic facility 99
Using the CEDF transaction 99
Command syntax 99
Command options 99
Beginning an EDF session (single-screen mode) 100
Beginning an EDF session (dual-screen mode) 101
Monitoring remote transactions 101
Invoking CEBR and CECI from CEDF 101
Using the CEDX transaction 102
Command syntax 103
Command options. 103

Chapter 13. CEKL - master terminal (emergency use) 105
Minimum abbreviation of keywords. 105
CEKL INQUIRE TASK 105
CEKL SET TASK 108

Chapter 14. CEMN - CICS monitoring facility 111

Chapter 15. CEMT - master terminal 113
Starting the CEMT transaction. 113
System parameters 114
Program function (PF) keys. 114
Minimum abbreviation of keywords. 115
ALL option 115
CLASS option 115
Family of resources (using * and + symbols) . . 115
Subdefining resource groups 116
Overtyping a display 117
Tab key 117
Scrolling symbol (+ sign) 117
Blank fields in a display. 117
Using the ? key. 118
Expanded panels 118
Hexadecimal values 119
Variables 119
Invoking CEMT from an application program 120
Security checking 120
CEMT DISCARD commands 120
CEMT DISCARD 122
CEMT INQUIRE commands 127
CEMT INQUIRE ATOMSERVICE. 127
CEMT INQUIRE AUTINSTMODEL 131
CEMT INQUIRE AUTOINSTALL. 132
CEMT INQUIRE AUXTRACE 134
CEMT INQUIRE BEAN 137
CEMT INQUIRE BRFACILITY. 138
CEMT INQUIRE BUNDLE 142
CEMT INQUIRE CFDTPOOL 144
CEMT INQUIRE CLASSCACHE 146
CEMT INQUIRE CONNECTION. 148
CEMT INQUIRE CORBASERVER 158
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE DB2ENTRY 172
CEMT INQUIRE DB2TRAN 178
CEMT INQUIRE DELETSHIPPED 180
CEMT INQUIRE DISPATCHER 182
CEMT INQUIRE DJAR 185
CEMT INQUIRE DOCTEMPLATE 189
CEMT INQUIRE DSAS 192
CEMT INQUIRE DSNAME. 195
CEMT INQUIRE DUMPDS. 203
CEMT INQUIRE ENQ 204
CEMT INQUIRE ENQMODEL 208
CEMT INQUIRE EPADAPTER 212
CEMT INQUIRE EVENTBINDING 216
CEMT INQUIRE EVENTPROCESS 218
CEMT INQUIRE EXCI 219
CEMT INQUIRE FEPI 220
CEMT INQUIRE FILE 221
CEMT INQUIRE GTFTRACE 229
CEMT INQUIRE HOST 231
CEMT INQUIRE INTTRACE 232
CEMT INQUIRE IPCONN 233
CEMT INQUIRE IRBATCH. 241
CEMT INQUIRE IRC. 241
CEMT INQUIRE JMODEL 243
CEMT INQUIRE JOURNALNAME 246
CEMT INQUIRE JVM 248
CEMT INQUIRE JVMPOOL 250
CEMT INQUIRE JVMSERVER. 251
CEMT INQUIRE LIBRARY. 254
CEMT INQUIRE LINE 259
CEMT INQUIRE MODENAME 261
CEMT INQUIRE MONITOR 263
CEMT INQUIRE MQCONN 265
CEMT INQUIRE MQINI 269
CEMT INQUIRE NETNAME 271

CEMT INQUIRE PARTNER 277
CEMT INQUIRE PIPELINE 279
CEMT INQUIRE PROCESSTYPE 282
CEMT INQUIRE PROFILE 286
CEMT INQUIRE PROGRAM 288
CEMT INQUIRE REQUESTMODEL 297
CEMT INQUIRE RRMS 301
CEMT INQUIRE STATISTICS 302
CEMT INQUIRE STREAMNAME 305
CEMT INQUIRE SYDUMPCODE. 307
CEMT INQUIRE SYSTEM 310
CEMT INQUIRE TASK 317
CEMT INQUIRE TCLASS 322
CEMT INQUIRE TCPIP 325
CEMT INQUIRE TCPIPSERVICE 326
CEMT INQUIRE TDQUEUE 334
CEMT INQUIRE TEMPSTORAGE 341
CEMT INQUIRE TERMINAL 342
CEMT INQUIRE TRANSACTION 349
CEMT INQUIRE TRDUMPCODE 356
CEMT INQUIRE TSMODEL 359
CEMT INQUIRE TSPOOL 362
CEMT INQUIRE TSQUEUE / TSQNAME. . . . 364
CEMT INQUIRE UOW 368
CEMT INQUIRE UOWDSNFAIL 372
CEMT INQUIRE UOWENQ 377
CEMT INQUIRE UOWLINK 381
CEMT INQUIRE URIMAP 385
CEMT INQUIRE VTAM 391
CEMT INQUIRE WEB 394
CEMT INQUIRE WEBSERVICE 395
CEMT INQUIRE WORKREQUEST 401
CEMT INQUIRE XMLTRANSFORM. 405
CEMT PERFORM commands 408
CEMT PERFORM CLASSCACHE 409
CEMT PERFORM CORBASERVER 411
CEMT PERFORM DELETSHIPPED 413
CEMT PERFORM DJAR. 413
CEMT PERFORM DUMP|SNAP 414
CEMT PERFORM ENDAFFINITY 415
CEMT PERFORM JVMPOOL 416
CEMT PERFORM PIPELINE 417
CEMT PERFORM RESET 418
CEMT PERFORM SECURITY 418
CEMT PERFORM SHUTDOWN 419
CEMT PERFORM STATISTICS 420
CEMT SET commands 425
CEMT SET ATOMSERVICE. 426
CEMT SET AUTOINSTALL 427
CEMT SET AUXTRACE 428
CEMT SET BRFACILITY 430
CEMT SET BUNDLE 431
CEMT SET CLASSCACHE 431
CEMT SET CONNECTION. 432
CEMT SET CORBASERVER 438
CEMT SET DB2CONN 440
CEMT SET DB2ENTRY 448
CEMT SET DB2TRAN 452
CEMT SET DELETSHIPPED 453
CEMT SET DISPATCHER 454
CEMT SET DOCTEMPLATE 456
CEMT SET DSAS 457
CEMT SET DSNAME 458
CEMT SET DUMPDS. 462
CEMT SET ENQMODEL 463
CEMT SET EPADAPTER 464
CEMT SET EVENTBINDING 465
CEMT SET EVENTPROCESS 465
CEMT SET FEPI 466
CEMT SET FILE 466
CEMT SET GTFTRACE 471
CEMT SET HOST 472
CEMT SET INTTRACE 472
CEMT SET IPCONN 473
CEMT SET IRC. 477
CEMT SET JOURNALNAME 478
CEMT SET JVMPOOL 479
CEMT SET JVMSERVER. 480
CEMT SET LINE 482
CEMT SET MODENAME 483
CEMT SET MONITOR 484
CEMT SET MQCONN 488
CEMT SET NETNAME 490
CEMT SET PIPELINE 492
CEMT SET PROCESSTYPE 493
CEMT SET PROGRAM 495
CEMT SET QUEUE 498
CEMT SET STATISTICS 498
CEMT SET SYDUMPCODE 501
CEMT SET SYSTEM 503
CEMT SET TASK 507
CEMT SET TCLASS 508
CEMT SET TCPIP 509
CEMT SET TCPIPSERVICE 511
CEMT SET TDQUEUE 512
CEMT SET TEMPSTORAGE 514
CEMT SET TERMINAL 515
CEMT SET TRANSACTION 519
CEMT SET TRDUMPCODE 520
CEMT SET TSQUEUE / TSQNAME. 522
CEMT SET UOW 523
CEMT SET UOWLINK 523
CEMT SET URIMAP 524
CEMT SET VTAM. 525
CEMT SET WEB 527
CEMT SET WEBSERVICE 528
CEMT SET WORKREQUEST 529
CEMT SET XMLTRANSFORM 530

Chapter 16. CEOT - terminal status 533
Using CEOT in inquiry mode 533
CEOT in set mode. 534

Chapter 17. CESF - sign off 537

Chapter 18. CESL - sign-on long . . . 539
CESL: IBM 3270 family display devices. 540
CESL: Non-3270 family terminals. 541
Using CESL when already signed-on 543

Chapter 19. CESN - sign on 545

CESN: IBM 3270 family display devices 546
CESN: Non-3270 family terminal 547
Using CESN when already signed-on 548
Changing CESN to permit password phrases . . . 548

Chapter 20. CEST - supervisory terminal 551

Chapter 21. CETR - trace control . . . 553
Defining tracing levels by component 554
Specifying transaction and terminal-related trace options 555
Specifying trace options for pooled JVMs in CETR 557

Chapter 22. CIDP - inactivate debugging profiles 559

Chapter 23. CIND - indoubt testing tool. 561
Running the CIND indoubt tool 562
Inquiring on the status of CIND 563
Running the indoubt tool using an EXEC CICS LINK command 563
Setting a single transaction indoubt 563
CIND Messages 563

Chapter 24. CLDM - CICS load module map 565

Chapter 25. CLER - Language Environment runtime options 567

Chapter 26. CMAC - messages and codes display 569

Chapter 27. CMSG - message switching 573
Sending a message 573
Canceling a message 575
Message-switching continuation 575
Message delivery 576
Subsystems and terminal systems 577
CMSG command 578
CMSG command options 578
Message-switching responses 590
CMSG examples 597

Chapter 28. CPIA - unit of work resynchronization transaction 599

Chapter 29. CREA - create REQUESTMODELs for enterprise beans. 601
Function: 601
Using CREA. 602
Starting CREA 602
TranID Specification 603
REQUESTMODEL Specification 607
Results of using CREA 610
CREC 611

Chapter 30. CRTE - remote transactions 613

Chapter 31. CSFE - terminal and system test 615
Using CSFE terminal test 615
Tracing with CSFE. 616
Terminal builder trace 616
Using CSFE debug functions 616
Storage freeze 616
Storage violation trap. 617
Global trap/trace program exit and global kernel exit 618

Chapter 32. CSPG - page retrieval . . 619
Retrieve a page. 619
Single keystroke retrieval (SKR) 621
Page retrieval for logical devices 621
Page retrieval and partitions 622
Display information about logical messages . . 622
Message chaining with CSPG 622
CSPG message termination 625
Copying pages with CSPG 626
Using Autopage with CSPG 627
Displaying routed pages with CSPG. 627

Chapter 33. CWTO - write to console operator 629

Chapter 34. DSNC - CICS DB2 transaction DSNC 631

Appendix. List of CICS transactions 633

Notices 643
Trademarks 644

Bibliography 645
CICS books for CICS Transaction Server for z/OS 645
CICSPlex SM books for CICS Transaction Server for z/OS 646
Other CICS publications. 646
Other IBM publications 646

Accessibility 649

Index 651

Preface

What this book is about

This book tells you how to start the transactions supplied with CICS®. It tells you how to control the CICS system, how to send messages, how to collect statistics, and how to get terminal error information. It is also useful if you are responsible for preparing procedures for people who operate such terminals.

The book also describes how you can use the operating system console as a CICS terminal.

Who should read this book

This book is intended for CICS terminal operators and persons responsible for preparing terminal operating procedures. It is also useful for both system and application programmers.

What you need to know to understand this book

A general knowledge of the concepts and terminology is required.

How to use this book

This book is mainly for reference. After a short introduction, the rest of the book is arranged in alphabetic order of the transaction identifier. An index is provided to pinpoint individual transactions and keywords.

Some CICS transactions are described in greater detail in other CICS books. These include:

Transaction	**CICS book**
CBAM	*CICS Business Transaction Services*
CDBC	*CICS IMS Database Control Guide*
CDBI	*CICS Data Areas*
CEBR	*CICS Application Programming Guide*
CECI	*CICS Application Programming Guide*
CECS	*CICS Application Programming Guide*
CEDA	*CICS Resource Definition Guide*
CETR	*CICS Problem Determination Guide*
CRTE	*CICS RACF Security Guide*
CSFE	*CICS Problem Determination Guide*
DSNC	*CICS DB2 Guide*

In these cases, this book contains a brief description of the transaction and indicates where further information is to be found.

© Copyright IBM Corp. 1982, 2011

Front end programming interface

Information about the Front End Programming Interface (FEPI) feature is not given in this book. See the *CICS Front End Programming Interface User's Guide* for relevant information.

Notes on terminology

- CICS refers to IBM® CICS Transaction Server for z/OS®, Version 4 Release 2
- VTAM® refers to IBM ACF/VTAM
- IMS™ refers to IBM IMS

Changes in CICS Transaction Server for z/OS, Version 4 Release 2

For information about changes that have been made in this release, please refer to *What's New* in the information center, or the following publications:

- *CICS Transaction Server for z/OS What's New*
- *CICS Transaction Server for z/OS Upgrading from CICS TS Version 4.1*
- *CICS Transaction Server for z/OS Upgrading from CICS TS Version 3.2*
- *CICS Transaction Server for z/OS Upgrading from CICS TS Version 3.1*

Any technical changes that are made to the text after release are indicated by a vertical bar (|) to the left of each new or changed line of information.

© Copyright IBM Corp. 1982, 2011

Chapter 1. Using CICS supplied transactions

CICS provides operations, usually initiated from terminals, called *transactions*, each of which involves the use of CICS tables, programs, and internal services. The following sections describe the transactions that are supplied by CICS and that have an operator interface.

CICS transactions have identification codes that start with "C" and are 4 characters long; for example, CEMT. For a complete list of CICS transactions, including those that do not have an operator interface, see the "List of CICS transactions," on page 633.

In general, you start a CICS transaction by entering its transaction identifier; for example, CEMT. The transaction identifier is used by CICS to identify the programs that handle the specified transactions, and to establish a task to process them.

If you use an IBM 3270 system display or similar display device that has the appropriate features installed, you can also start a transaction by a program function (PF) key or program attention (PA) key, by an operator identification card reader, by a magnetic slot reader, or by a light pen.

You might want to apply a CICS-supplied upgrade, but are using modified versions of one or more CICS-supplied transactions or of the CICS-supplied calling programs that handle CICS-supplied transactions. After you have first copied them to differently named groups, you must replace these private versions from the upgraded CICS-supplied version and modify them afresh to ensure that the necessary upgrade changes are carried out. Failure to do this can lead to unpredictable results.

This chapter describes:

- "How to start or stop a transaction"
- "Syntax notation and conventions used" on page 2
- "Terminal operation" on page 4
- "Using the system console" on page 9
- "Using TSO consoles" on page 11

How to start or stop a transaction

You start a CICS transaction by pressing the CLEAR key to clear the screen, and entering the transaction identifier, either by itself or followed by data, on the command line of the screen. The command line is a single line, usually at the top of the screen.

About this task

You can type the transaction identifier by itself and follow the prompts until a complete transaction command is built up, or you can type the complete transaction command on the command line. If you do not enter enough information, or if the information you enter is wrong, you are prompted to complete or correct your input.

© Copyright IBM Corp. 1982, 2011

For example, in the following transaction, CEMT is the transaction identifier and the additional data is INQUIRE PROGRAM(PROG1).

```
CEMT INQUIRE PROGRAM(PROG1)
```

When the transaction starts, it processes the additional data. At the completion of this transaction, you get the following message:

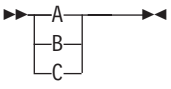
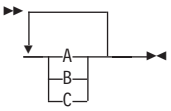
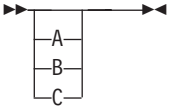
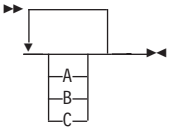
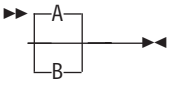
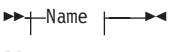
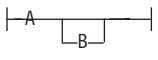
```
STATUS: SESSION ENDED
```

After a transaction has completed, press the CLEAR key to clear the screen in readiness for the next transaction. You can cancel any request by typing CANCEL on the command line.

Syntax notation and conventions used

Each command has a syntax box to show you what options there are. You interpret the syntax by following the arrows from left to right.

The conventions are:

Symbol	Action
	A set of alternatives—one of which you ***must*** code.
	A set of alternatives—one of which you ***must*** code. You ***can*** code more than one of them, in any sequence.
	A set of alternatives—one of which you ***can*** code.
	A set of alternatives — any number (including none) of which you can code once, in any sequence.
	Alternatives where **A** is the default.
 **Name:** 	Use with the named section in place of its name.

Symbol	Action
Punctuation and uppercase characters	Code exactly as shown.
Lowercase characters	Code your own text, as appropriate (for example, `name`).

Minimum abbreviation of keywords

In general, the CICS transactions accept as few characters of a keyword as needed to identify it uniquely in the request.

For example, in a CEMT INQUIRE TASK command, you can use TASK, TAS, or TA to uniquely identify TASK. You cannot use T alone because that could be confused with TCLASS, TERMINAL, TRACE, or TRANSACTION.

In the syntax displays on your screen, the minimum permitted abbreviation is shown in uppercase characters, and the remainder is shown in lowercase.

Minimum abbreviations might change between CICS releases because of the introduction of new commands.

Uppercase input to transactions

In general, most CICS-supplied transactions accept only uppercase input. If UCTRAN=YES has been specified in the terminal definition, all lowercase characters, even those enclosed within single quotation marks, are translated to uppercase.

If you have to specify UCTRAN=NO for your terminal, you have to ensure that the group specified for your terminal refers to a profile that will carry out uppercase translation.

CICS provides a PROFILE definition, DFHCICSP, in the DFHSTAND group in the CICS system definition (CSD) file. This profile is identical to DFHCICST except that it specifies UCTRAN(YES) instead of UCTRAN(NO).

The new profile is used by the CICS-supplied page retrieval transaction, CSPG. The new profile, together with changes in the task-attach routine and the page retrieval program, enables CICS to perform uppercase translation at the transaction level for BMS paging.

This allows users of terminals that are defined with uppercase translation switched off to use the page retrieval function without having to enter paging commands in upper case. Assigning a new profile for CSPG means that all data entered on the retrieval command (defined by the PGRET system initialization parameter) and the purge command (defined by the PGPURGE system initialization parameter) is translated to uppercase.

If a user's terminal is defined with UCTRAN(YES), the new profile has no effect because all terminal input is translated to uppercase anyway.

Terminal operation

A CICS system makes provision for the following classes of operators; terminal operator, a supervisory terminal operator, and a master terminal operator.

- A *terminal operator* who can use a terminal to perform routine transactions that cause application programs to be processed. You can use a small selection of CICS transactions. For example, you can inquire about, or change, the status of your own terminal.
- A *supervisory terminal operator* who can perform all the duties of a terminal operator, in addition to supervising other operators within a functional group. Your operator security code gives you access to the supervisory terminal transaction, CEST, with which you can monitor and control some of the system resources used by your group.
- A *master terminal operator* who can monitor and control resources in a CICS system. Your operator security code gives you access to the master terminal transaction, CEMT, with which you monitor and control the system resources. Internal security checking might limit the range of resources under your control.

$ (the dollar symbol)

In the character sets given in this information, the dollar symbol ($) is used as a national currency symbol and is assumed to be assigned the EBCDIC code point X'5B'.

In some countries a different currency symbol, for example the pound symbol (£), or the yen symbol (¥), is assigned the same EBCDIC code point. In these countries, the appropriate currency symbol should be used instead of the dollar symbol.

Operator security

The transactions you can initiate are defined by your profile in the external security manager (ESM) database, which is normally provided when you sign on using the CESN transaction.

Generally, the master terminal operator has access to all CICS-supplied transactions, the supervisory terminal operator has access to a subset, and the terminal operator has access to very few transactions.

The system programmer is responsible for allocating operator security codes to restrict the use of particular transactions. For more information, see the relevant system administration guide for the ESM you are using. For example, if you are using RACF, see the *z/OS Security Server RACF Security Administrator's Guide*.

Terminal operator

To operate the system, you normally first sign on to the system and, as a minimum, enter your user ID and your password.

During signon, the information you enter is used by CICS to establish priorities and your ESM profile for the transactions that you might want to use later. When you have signed on, you have access to those transactions defined in your ESM profile.

After you have signed on, you can enter only specific transaction identifiers. Be aware of error messages that might be generated by the transactions you start, and the corrective action that you must take. In addition to error messages, be aware of other messages that CICS might transmit to your terminal.

You need to know the terminal identifiers of other terminals with which you want to communicate.

Supervisory terminal operator

A supervisory terminal operator is the supervisor of any part of the system for which group control is desired. You are responsible for supervising, and keeping operational, groups of terminals defined in one or more terminal list tables (TLTs). You do this using the supervisory terminal transaction, CEST.

Your responsibilities can be thought of as a subset of those of the master terminal operator, and you should be aware of those functions that are not available to the terminals under your supervision. You also should be aware of, and understand, the procedure for changing the status of each terminal.

You should know the identifiers of all terminals and operators under your supervision. The terminal identifiers are defined in one or more CICS terminal list tables (TLTs). Individual TLTs can be identified by a 1-or 2-character suffix you enter as the SUPRID and CLASS(value) keywords of the CEST transaction.

When you use the CEST command for *all* terminals defined in a TLT, you have to specify the SUPRID keyword followed by the 2-character identifier of that TLT.

A subset of those terminals can be grouped together as a *class*, and can be defined as such in a different TLT. For information about defining the TLT itself, see the *CICS Resource Definition Guide*. You can specify that class by means of the CLASS(value) keyword of the CEST SET TERMINAL command, where 'value' is the suffix that identifies the TLT in which the class of terminals has been defined.

Alternatively, you can name one or more terminals in the CEST SET TERMINAL(value) command itself.

If you frequently want to restrict a CEST command to a subset of your terminals, and have defined another TLT identifying that subset, you then have to use the CEST SET TERMINAL SUPRID(value) keyword to refer to the "main" TLT, followed by the CLASS(value) keyword to refer to the specific TLT containing the subset.

So, for example, if you have defined terminals S202, S203, S204, and S205 in DFHTLTAB and you want to issue a CEST command that sets *all* those terminals out of service, you issue the following command:

```
CEST SET TERMINAL SUPRID(AB) OUTSERVICE
```

If, on the other hand, you want to restrict your command(s) to terminals S202 and S204, for example, you could define these two terminals in another TLT—DFHTLTCD, say—and issue the following command:

```
CEST SET TERMINAL SUPRID(AB) CLASS(CD) OUTSERVICE
```

Alternatively, of course, you could issue the following command:

```
CEST SET TERMINAL(S202,S204) SUPRID(AB) OUTSERVICE
```

Unless otherwise stated, the information in this book about the supervisory terminal and the CEST transaction applies only to a single CICS system, regardless of whether it is connected to another CICS system through ISC or MRO.

Master terminal operator

The master terminal operator controls system components using the master terminal transaction, CEMT. With this transaction, the master terminal operator can dynamically change the system control parameters.

Although the transaction can be started at any valid IBM 3270 family display device or equivalent terminal, or from the operating system console, its use is intended to be limited to a person known as the *master terminal operator*. Starting a transaction from the operating system console is described in "Using the system console" on page 9.

The control permitted through CEMT allows you, the master terminal operator, to improve performance by changing the system control parameters in the day-to-day operation of the system. In addition to system control, you have prime responsibility for administering the terminal facilities of the system.

By using the routing transaction (CRTE), you can also be a master terminal operator for multiple connected CICS systems.

As the master terminal operator, you can access all terminal and supervisory terminal transactions. In addition, however, you must be familiar with all the procedures associated exclusively with the master terminal. You must be aware of which terminals and operators can access CICS at any given time, and of the identifiers by which they are known to CICS.

For example, when inquiring about terminals, you can specify a class of terminals or a list of terminals. A class of terminals is specified by the CLASS(value) keyword, where 'value' is the 1-or 2-character suffix of the related terminal list table (DFHTLTxx). A list of terminals is specified by a series of terminal identifiers following the CEMT SET TERMINAL(t1,t2,...,) command, where t1, t2, are terminal identifiers. See "List of resource identifiers" on page 116 for more information.

For MRO and LUTYPE6.1 connections, you must know the identifier of each parallel session, and specify this identifier when operating on the session.

For LUTYPE6.2 (APPC) connections, you must know the modename of each set of parallel sessions, and specify this modename when operating on the modegroup.

Your use of the master terminal transaction is restricted by entries in the signon table and in the installed transaction definitions. These entries are the responsibility of the system programmer.

During long periods of continuous operation, you can, at intervals, read out and reset the statistics counters. The volume of activity in your system determines how often you should do this.

When the system has satisfactorily completed its response to a command, the time and date are printed or displayed at your terminal, as follows:

```
TIME=hh.mm.ss DATE=mm.dd.yy
```

where time is in hours, minutes, and seconds, and date is in months, days, and years, or in the form specified by the DATFORM system initialization parameter. For brevity this final message has been deleted from all further examples.

Unless otherwise stated, the information about the master terminal and its transactions applies only to a single CICS system, regardless of whether it is connected to another CICS system through ISC or MRO.

MRO and ISC support

Multiregion operation (MRO) and intersystem communication (ISC) allow the sharing of resources between more than one CICS region. Thus a user at a terminal assigned to one CICS region can run transactions in connected regions, and can access resources - files, for example - in other regions.

It is also possible for a transaction running in one region to communicate with a transaction running in another region, thus sharing the processing workload.

Except for experiencing longer response times, you should not be aware that MRO or ISC processes are being used.

BMS partitions

When you use display devices that support BMS partitions, make sure that you understand how to use the SCROLL, PARTITION JUMP, CLEAR, and CLEAR PARTITION keys, the concept of the active partition and the meanings of the partition-related indicator symbols that can appear on a display screen.

For information about BMS partitions, see the *CICS Application Programming Guide*.

CLEAR key

The CLEAR key clears all partitions from the display, and sets the terminal to 'base' state. The next BMS output request recreates the partitions (but does not restore their contents), using the application partition set.

The CLEAR and CLEAR PARTITION keys cannot be used interchangeably when an existing CICS transaction is run in a single explicit partition.

Partitions and the execution diagnostic facility

The execution diagnostic facility (EDF), invoked by CEDF, is unavailable in single-screen mode on a terminal in partitioned state. EDF must be used in dual-screen mode for debugging application programs that use partitions.

Partitions and the command interpreter

The CICS command interpreter, invoked by CECI or CECS, cannot be used to process commands that refer to partitions. This is because the command interpreter display cannot be restored after the screen has been partitioned.

PA1 print key

The PA1 print key is not supported from a terminal in partitioned state.

Routing and multiple partitions

Routed messages can be directed to a terminal, including the transaction terminal, which supports partitions. However, such messages reset the terminal to 'base' state.

Terminal paging

When a BMS logical message is saved in CICS temporary storage, CICS also saves the application partition set. This partition set is loaded onto the target terminal before any pages are delivered. CICS builds a separate page for each partition, and overflow occurs on a partition basis.

Page retrieval

Terminal-operator page-retrieval commands operate on a partition basis. When a page-retrieval or page-copy command is entered in a partition, it implicitly refers to pages in that partition. If single-keystroke retrieval is used, the retrieval command applies to the partition containing the cursor when the PF key is pressed. The first page for a partition is displayed initially in the viewport.

Message chaining

CICS retains a current partition for each level of page chaining. This is initially the default partition for that partition set. Page-retrieval commands entered on a cleared screen, or page-retrieval commands for a chaining level other than the one being displayed, refer to the current partition for the target chaining level. The current partition is reset to the partition in which the last terminal-operator command was entered.

CICS retains the current page for each partition in the partition set. This is initially the first page. The current page is redisplayed in each partition in the following circumstances:

- For the initial display when the BMS paging program is first invoked
- Following erasure of the terminal partition set caused by pressing the CLEAR key
- Following page retrieval for a different page-chaining level
- Following page purge for a different page-chaining level.

Copying pages

BMS page copy operates on a partition basis (not a screen or partition set basis). BMS page copy copies a page from a partition to any terminal in 'base' state. You cannot copy a page from a partition to another partition on the same or another terminal.

Message termination

When you terminate a message, the entire logical message (that is, all pages in all partitions) is purged, irrespective of the partition in which you entered the purge command. The response to a page query request is displayed on a cleared, unpartitioned screen.

Error messages

Most error messages relating to invalid paging commands are displayed with an erase or write in the partition in which you entered the command. Other error messages unrelated to any particular partition (such as those relating to invalid message identifiers) are displayed on a cleared unpartitioned screen.

Using the system console

Console support makes it possible for a terminal to be both an operating system console and a CICS master terminal.

About this task

If multiple console support (MCS) is in use, you can define each console to CICS as a separate terminal, and all consoles can communicate with CICS simultaneously.

You can use any operating system console as a CICS terminal, if it has been specified on the CONSOLE keyword of the **CEDA DEFINE TYPETERM** command. If this has not been done, you get the following message when you try to use the console:

```
DFHAC2015 This console has not been defined to CICS.
```

and your input is ignored.

All consoles that have been defined as CICS terminals can use automatic transaction initiation (ATI), and can receive messages from other terminals and consoles, as well as from CICS transactions.

In a system that has consoles and SNA LUs, a console can remain active when CICS and the z/OS Communications Server are disconnected from each other. You can use the console to make or break the CICS-z/OS Communications Server connection without CICS being terminated.

Use the MODIFY and REPLY commands to start the CICS-supplied transactions from an operating system console.

In addition to the MODIFY and REPLY commands, the system programmer should consider use of the CONTROL, DISPLAY, START, and VARY commands when preparing console operator procedures. For information on these commands and other system details, see the *z/OS MVS System Commands* manual.

Rules for console entry

Commands typed at a console are translated to uppercase, except for characters enclosed within single quotation marks (' '), which remain unchanged. The occurrence of a literal single quotation mark must be indicated by a pair of single quotation marks (''), for example:

```
'Please phone Mr O''Neill'.
```

If UCTRAN=YES has been specified in the terminal definition, all lowercase characters, even those enclosed within single quotation marks, are translated to uppercase.

MODIFY command

You start a CICS transaction from a console by using the MODIFY command, as follows:

```
MODIFY ident,datastring
```

You can abbreviate the MODIFY command to F.

`ident` can be any of the following:

- The name of the job used to start CICS, when it is started by a job stream.
- The name of the procedure used to start CICS, when it is started by an MVS™ START command, for example:

  ```
  START procedure_name
  ```

  where “procedure_name” is the ident value.
- The task identifier that was used to qualify the procedure name, for example:

  ```
  START procedure_name.taskid
  ```

  where “taskid” is the ident value. This is likely to be used where the same procedure is started more than once.

`datastring` is a string of data, starting with a CICS transaction identifier.

For example, to start transaction CEBT on the CICSA system from the console, type:

```
MODIFY CICSA,CEBT PERFORM TAKEOVER
```

You can type more than one MODIFY command at a console; each is processed in order of entry.

A CICS transaction can issue terminal control READ, WRITE, or CONVERSE commands to communicate with a console operator. WRITE and CONVERSE transmit application program messages, but READ produces a prompt, incorporating message ‘DFH4200A', as follows:

```
@nn DFH4200A jjjjjjjj tttt
```

where:

nn is the number (generated by the operating system) that you must use in your reply to the prompt. Messages from a transaction that uses CONVERSE commands also contain this number.

jjjjjjjj is the jobname of CICS in the operating system.

tttt is the transaction identifier of the CICS transaction that has issued the READ command.

REPLY command

You (the console operator) must respond to each prompt by using the REPLY command, which you can type at either the prompted console or the master console:

```
R[EPLY] nn,datastring
```

where `nn` is the number of the prompt to which you are replying, and `datastring` is your reply.

If a transaction is purged while it is awaiting a reply from the operator, the reply is canceled.

You should note that messages to the console can become interspersed with messages from the operating system and from other regions, making them difficult to read. In extreme cases, parts of lengthy messages can 'scroll off' the console screen before they have been read.

Example of a conversation using CONVERSE

```
modify job002,serv  1

@17 FAULT TYPE?   2

r 17,elec   3

MESSAGE HAS BEEN SENT
```

1 MODIFY command specifying that transaction "serv" is to be started; this transaction sends messages to service groups supporting the installation.

2 The transaction response produced by a CONVERSE command and relayed by the operating system.

3 Your reply that the fault is an electrical one.

Example of a conversation using WRITE/READ

```
modify job002,usid   1

USER SIGNON ID=?   2

@25 DFH4200A JOB002 USID  3

r 25,accts1   4

USER'S NAME: J. SMITH   5
USER'S TEL. NO.: 88999   6
```

1 MODIFY command specifying that transaction "usid" is to be started. This transaction provides information about the user identified by "usid".

2 Application-program message produced by a WRITE command.

3 System message produced by a READ command.

4 Your reply.

5 Transaction message, produced by a WRITE command, giving the requested information. No reply is needed.

6 Another transaction message, produced by a WRITE command, giving more requested information. Again, no reply is needed.

Using TSO consoles

A TSO session can be used to input CICS commands. This has several advantages as it removes the MVS limitation of 99 consoles, it supports additional device types such as the IBM 3290, and it supports remote operation.

About this task

The console ID is 4 bytes, only one of which is used for locally connected consoles. TSO and JES3 consoles use all 4 bytes. In addition, each console in a sysplex has an 8-byte name.

Using the KILL option for purging transactions

It is possible for a looping CICS task or shortage of CICS resources to bring all normal processing in a CICS region to a halt. The KILL option helps you to remove such tasks more quickly from a CICS region, with the minimum impact on the integrity of your systems.

About this task

You must use the KILL option only when it becomes impossible for you to cancel the task or tasks using any of the standard CICS mechanisms. The KILL option does not guarantee integrity of any kind, but in some situations allows you to free a stalled region, enabling the region to continue processing.

Procedure

1. Try to cancel the task using the PURGE or FORCEPURGE option on the **CEMT SET TASK** command. Both PURGE and FORCEPURGE could enable you to remove the task, retaining system and data integrity. If neither PURGE or FORCEPURGE respond within a reasonable time, use the KILL option.
2. Use the KILL option on the appropriate command, depending on what task you are trying to cancel. The KILL option is available on **CEMT SET TASK**, **CEMT SET CONNECTION**, and **CEMT SET TERMINAL** commands.
3. If you cannot list the active tasks in the system to determine which task or tasks should be canceled, use the CEKL transaction.
   a. Enter the command `CEKL INQUIRE TASK` to list information about tasks in the CICS region; for example, all tasks that are suspended.
   b. Enter the command `CEKL SET TASK` to cancel a selected task using the PURGE, FORCEPURGE or KILL option.

Using CEMT to terminate a task

Having identified the problem task, you can then use the normal process of attempting to remove the task from your system using either PURGE or FORCEPURGE through the CEMT SET TASK (value) PURGE transaction.

About this task

Initially you can track the state of the task to see if it has successfully removed. Two things may have happened:

- The task was not purged within an acceptable length of time or
- The CEMT transaction did not respond.

If the task could not be purged but CEMT is available, try using the new KILL option. You must try FORCEPURGE before using KILL. See “Consequences of using KILL” on page 13 before using this option.

If CEMT is unavailable, consider using the new CEKL command, see “Using the console to terminate a task” on page 13.

Using the console to terminate a task

You can identify problems in your system using the CEMT transaction or CICSPLEX SM. However, if these are unavailable or not running (because the QR TCB is looping or suspended), you can use the new command, CEKL.

About this task

You can use the CEKL command to do the following:

- To identify the problem tasks
- To remove the tasks by executing the PURGE, FORCEPURGE, or KILL option
- To monitor the effects of attempting to remove the tasks from your system.

For full details about CEKL, see Chapter 13, "CEKL - master terminal (emergency use)," on page 105.

Identifying tasks using CEKL

You can use CEKL INQUIRE TASK in a number of ways:

- CEKL INQUIRE TASK returns a list of tasks on your system, see Chapter 13, "CEKL - master terminal (emergency use)," on page 105 for more information.
- You can restrict the list of tasks by specifying the TRANCLASS or TRANSID option.
- You can also restrict the list by specifying: DISPATCHABLE, or RUNNING, or SUSPENDED, for example, using DISPATCHABLE returns all the dispatchable tasks in your system.

Removing tasks using CEKL

CEKL SET TASK(value) has three options: PURGE, FORCEPURGE or KILL. The preferred sequence of issuing the commands is as follows:

1. Try PURGE first if the task is purgeable.
2. If the task is not purged in a reasonable time, try using the FORCEPURGE option.
3. If the task is not purged in a reasonable time, try using the KILL option.

Note: Using CEKL you can request a task to be killed without having first tried PURGE or FORCEPURGE, see "Consequences of using KILL."

Monitoring tasks using CEKL

CEKL INQUIRE TASK(*value*) returns the details of a single task. This is useful to monitor a task after you have requested it to be purged, forcepurged or killed.

Consequences of using KILL

WARNING: This action can result in unpredictable effects, including overwriting of data within the CICS region or abnormal termination of the CICS region. It should only be used as a last resort.

Following the use of the KILL option, an abend code is issued that reports the protective state of the task (purge or forcepurge protection) when it was killed. Regardless of the state of the transaction KILL cannot guarantee any data or system integrity.

Using CEMT (or equivalent SPI commands), you are not allowed to kill a task without having first attempted to forcepurge it.

You can use CEKL to kill the task without having first forcepurged it. Use CEKL is very much the option of last resort.

Removing tasks can have an effect on the integrity of your system, Table 1 shows the possible consequences of using purge, forcepurge, or kill.

Table 1. Data and system integrity

	System integrity	Data integrity
PURGE	YES	YES
FORCEPURGE	YES	NO
KILL	NO	NO

CICS ignores a request to purge a task if the task is marked non-purgeable.

CICS might ignore or delay processing a request because of a risk of damaging CICS system integrity, for example, because I/O might still be in progress to a buffer owned by the task, or because recovery processing would be unable to complete.

Chapter 2. The application debugging profile manager

Use the application debugging profile manager to manage your debugging profiles.

You can use it to perform the following functions:

- Display a list of debugging profiles
- View the contents of profiles
- Create new profiles
- Modify existing profiles
- Copying existing profiles
- Delete debugging profiles
- Activate and inactivate profiles
- Associate a debugging display device with active profiles

The application debugging profile manager has two user interfaces:

- A Web browser interface. For more information, see "Using the application debugging profile manager Web interface."
- A terminal interface (the CADP transaction). For more information, see "Using the application debugging profile manager 3270 interface" on page 37.

A utility transaction (CIDP) lets you inactivate all debugging profiles in the system. For details see Chapter 22, "CIDP - inactivate debugging profiles," on page 559.

Using the application debugging profile manager Web interface

Start the application debugging profile manager by typing its URL in your Web browser. The URL that you enter will depend upon how CICS Web support is configured. Your system administrator will tell you the URL to use for your system. CICS displays the "List profiles" page.

About this task

For example, if your Web browser connects directly to CICS, and your system is configured to use the sample analyzer program DFHWBADX, the URL is:

`http://mvs_address:port/CICS/CWBA/dfhdpwb`

If you are using URIMAP definitions to manage requests, the URL is as defined in the URIMAP definition that references program DFHDPWB.

Configuring access to the application debugging profile manager Web interface

To provide access to the Web interface, you can use either an analyzer program, or URIMAP definitions, as part of your CICS Web support architecture.

About this task

The *CICS Internet Guide* explains the architecture elements that are used to give Web clients access to CICS applications.

© Copyright IBM Corp. 1982, 2011

Only one URL is required. The user accesses different pages by selecting interface elements on the pages that the program provides, beginning with the “List profiles” page.

Procedure

- If you want to use the sample analyzer program DFHWBADX, or an analyzer program with similar function, to handle requests for the application debugging profile manager, configure the path component of the URL to give CICS the information it needs to run the program. The path `/CICS/CWBA/dfhdpwb` tells the analyzer that:
  - No converter program is required (`CICS` indicates this).
  - The default transaction ID CWBA is used.
  - The program DFHDPWB, which is the application debugging profile manager, is to be run.
- If you want to use URIMAP definitions to handle requests for the application debugging profile manager, you need to set up two URIMAP definitions.
  - One URIMAP definition matches the URL that is entered by the user to start the application debugging profile manager's Web interface, and maps to the program DFHDPWB.
  - The second URIMAP definition is used to run the CICS program DFHADWB1, which displays graphics.

  The second URIMAP definition is required because when a graphic is wanted, the HTML pages for the Web interface use the tag `img src=`, so that the Web browser makes a request for the graphic and then displays what it receives. The `img src=` tag specifies the name of the graphics program, and the Web browser makes the request using the path component of the URL that was originally used to start the Web interface, followed by the name of the graphics program. A URIMAP definition is needed to map this new request to the graphics program.

Example

The following sample URIMAP definitions could be used to provide access to DFHDPWB and DFHADWB1:

```
For DFHDPWB:

URIMAP:       ADPM               - URIMAP name
GROUP:        MYGROUP            - any suitable
USAGE:        SERVER             - For inbound requests
SCHEME:       HTTP               - Or use HTTPS for SSL
HOST:         *                  - * matches any host name
PATH:         /cicsapps/adpm/*   - Any path can be used, with final asterisk
TCPIPSERVICE:                    - If blank, applies to all ports
ANALYZER:     NO                 - Means do not run analyzer
CONVERTER:    CICS               - Means no converter used
TRANSACTION:  CWBA               - Default transaction ID
PROGRAM:      DFHDPWB            - App debugging profile manager

For DFHADWB1:

URIMAP:       ADWB1
GROUP:        MYGROUP
USAGE:        SERVER
SCHEME:       HTTP
HOST:         *
PATH:         /cicsapps/adpm/dfhadwb1
[Same path as for DFHDPWB, but with dfhadwb1 appended]
```

```
TCPIPSERVICE:
ANALYZER:     NO
CONVERTER:    CICS
TRANSACTION:  CWBA
PROGRAM:      DFHADWB1            - Graphics program
```

When URIMAP definitions are used, the path specified in the URL does not need to have any relationship to the resource; the linkage between the path and the resource is made by the URIMAP definition. For the URIMAP that points to DFHDPWB, you could specify a path that includes a page element, such as `/cicsapps/adpm/start.html`. The name of the page element does not matter. Note that if you do this, when the Web browser makes a request for DFHADWB1, it omits the page element and appends `dfhadwb1` in its place. If the path in the first of the sample URIMAP definitions was `/cicsapps/adpm/start.html`, the path in the second sample URIMAP definition would still be `/cicsapps/adpm/dfhadwb1`.

The *CICS Internet Guide* explains how to set up URIMAP definitions.

The List profiles page

Use the "List profiles" page to display a list of debugging profiles:

- When you start the application debugging profile manager's Web interface, CICS displays the "List profiles" page.

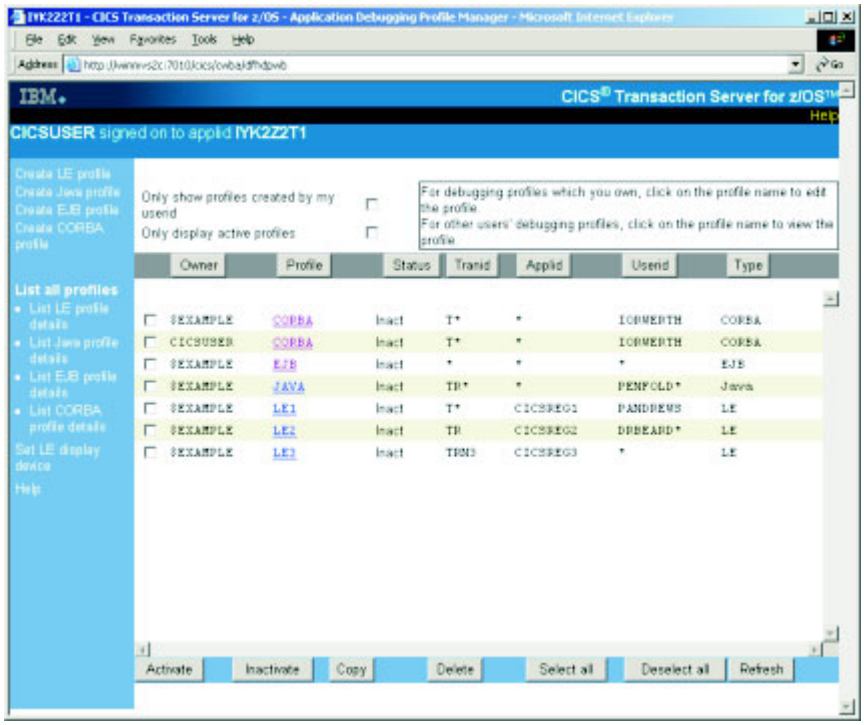


Figure 1. The List profiles page

When you use the debugging profile manager for the first time, CICS displays all profiles. Subsequently, CICS displays the profiles that were selected when you last used it.

If there are more profiles than can be displayed in the window, use the scrollbars to scroll backwards and forwards through the list. If you have no profiles, CICS displays an empty list.

There are four variants of the “List profiles” page:

List LE profiles
Lists only the profiles for compiled language profiles

List Java profiles
Lists only the profiles for Java programs

List EJB profiles
Lists only the profiles for enterprise beans

List CORBA profiles
Lists only the profiles for stateless CORBA objects

The behavior of these pages is identical to the “List profiles” page; however the information displayed on each is specific to the type of profile.

The list contains selected information from the debugging profile. The columns on the page are:

Owner
The userid of the profile owner; that is, of the user who created the profile.

Profile
The name of the profile

Status The status of the profile (**Act** for Active, or **Inact** for Inactive)

The following columns display information specified when the profile is created:

Tranid Displays the contents of the **transaction** field

Program
On the “List profiles” and “List LE profiles” pages only , displays the contents of the **program** field.

Compile Unit
On the “List profiles” and “List LE profiles” pages only, displays the contents of **Compile Unit** field.

If the Compile Unit name is too long to display in the available space, the leading characters are displayed, followed by “**...**”. To display the Compile Unit name in full, click on the profile name.

Applid
Displays the contents of the **Applid** field

Userid
Displays the contents of the **Userid** field

Termid
On the “List profiles” and “List LE profiles” pages only, displays the contents of the **Terminal** field.

Type On the “List profiles” page only, displays the type of program specified in the debugging profile:
CORBA
CORBA object
EJB Enterprise bean

Java Java program
LE Compiled language program

Netname
On the "List LE profiles" page only, displays the contents of the **Netname** field.

Class On the "List Java profiles" and "List CORBA profiles" pages only, displays the contents of the **Class** field.

If the Class name is too long to display in the available space, the trailing characters are displayed, preceded by "**...**". To display the Class name in full, click on the profile name.

Bean On the "List EJB profiles" page only, displays the contents of the **Bean** field.

If the bean name is too long to display in the available space, the leading characters are displayed, followed by "**...**" To display the bean name in full, click on the profile name.

Method
On the "List EJB profiles" and "List CORBA profiles" pages only, displays the contents of the **Method** field.

If the Method name is too long to display in the available space, the leading characters are displayed, followed by "**...**". To display the Method name in full, click on the profile name.

You can change the way information is displayed on the "List profiles" page:

Selecting which profiles are displayed
Use the checkboxes at the top of the page to select which debugging profiles are displayed. The options are:
- Display all profiles
- Display all profiles that you created
- Display all active profiles
- Display only active profiles that you created

Sorting the list
Use the buttons above each column to re-display the list in the sequence determined by the contents of the column. For example, to re-display the profiles in sequence of program name, click the **Program** button. CICS uses the EBCDIC sorting sequence when it re-displays the list.

Your choice of which profiles are displayed, and your chosen sequence, are preserved, and used the next time you use the debugging profile manager.

Related tasks

"Creating a debugging profile with the Web interface" on page 21
"Activating debugging profiles with the Web interface" on page 22
"Inactivating debugging profiles with the Web interface" on page 23
"Viewing a debugging profile with the Web interface" on page 23
"Changing a debugging profile with the Web interface" on page 24
"Copying debugging profiles with the Web interface" on page 24
"Deleting debugging profiles with the Web interface" on page 25
"Setting the display device" on page 34
"Combining actions on the List debugging profiles screen" on page 44

Related reference

"The Create compiled profile page" on page 26
"The Create Java profile page" on page 29
"The View LE profile page" on page 32
"The View Java profile page" on page 33
"The Set compiled display device page" on page 35

Buttons on the List profiles page

The buttons at the head of the following columns are used to re-display the list of profiles in sequence:

Owner
Profile
Status
Tranid
Program
Compile Unit
Applid
Userid
Termid
Netname
Type

The following buttons are inactive, and cannot be used to re-sequence the list of profiles:

Class

Bean

Method

Other actions are performed using the buttons at the bottom of the "List profiles" page:

Activate
Activate selected profiles. See "Activating debugging profiles with the Web interface" on page 22 for more information.

Inactivate
Inactivate selected profiles. See "Inactivating debugging profiles with the Web interface" on page 23 for more information.

Copy
Copy selected profiles. See "Copying debugging profiles with the Web interface" on page 24 for more information.

Delete
Delete selected profiles. See "Deleting debugging profiles with the Web interface" on page 25 for more information.

Select all
Selects all the profiles in the list.

Deselect all
Deselects all the profiles in the list

Refresh
Refresh the "List profiles" page. The list is updated to show any changes that you, and other users, have made.

Creating a debugging profile with the Web interface

About this task

You can create debugging profiles in these ways:
- You can create a completely new profile by entering all the information needed to define the profile
- You can base the new profile on an existing profile

Related reference

"The List profiles page" on page 17

Creating a new profile

About this task

Starting with the "List all profiles" page, follow these steps:

Procedure

- To create a profile for a compiled language (Language Environment®) program
  1. Click on **Create compiled profile**. CICS displays the "Create compiled profile" page.
  2. Complete the information that you need to specify your profile. See "The Create compiled profile page" on page 26 for details.
  3. Click on the **Create** button. CICS checks that you have entered valid data.
     - If your data is valid, the profile is saved
     - If your data contains an error, a message is displayed. Re-enter the data, and click on the **Create** button again.
  4. Click on **List all profiles** to return to the "List all profiles" page.
- To create a profile for a Java program
  1. Click on **Create Java profile**. CICS displays the "Create Java profile" page.
  2. Complete the fields that you need to specify your profile.
  3. Click on the **Create** button. CICS checks that you have entered valid data.
     - If your data is valid, the profile is saved
     - If your data contains an error, a message is displayed. Re-enter the data, and click on the **Create** button again.
  4. Click on **List all profiles** to return to the "List profiles" page.

Basing a new profile on an existing profile

About this task

You can create a new debugging profile using an existing profile as a starting point. The steps you take depend upon whether the original profile is owned by you or another user:

If you own the profile
Follow the steps described in "Changing a debugging profile with the Web interface" on page 24. Before you save the profile, give it a new name.

If another user owns the profile:
Follow the steps below.

Procedure

1. Copy the profile, following the steps described in "Copying debugging profiles with the Web interface" on page 24.
2. Make any changes to the profile by following the steps described in "Changing a debugging profile with the Web interface" on page 24.

Activating debugging profiles with the Web interface

About this task

To activate debugging profiles, start with the "List all profiles" page, and follow these steps:

Procedure

1. Use the check boxes at the top of the page to ensure that the display includes the profiles you want to activate.
2. Scroll the list to a profile that you want to activate.
3. Select the profile using the check box to the left of the profile name.
4. Repeat steps 2 through 3 to select all the profiles you want to activate.
5. Click the **Activate** button. By default, if any of the selected profiles is for a compiled language (Language Environment) program, CICS displays the "Set compiled display device" page.

   If none of the selected profiles is for a compiled language programs, CICS refreshes the "List all profiles" page.

   You can choose not to see the "Set compiled display device" page when you activate profiles. See "Setting the display device" on page 34 for more information.

Results

If you change a profile while it is active, the changes take effect immediately: the next time a program is started, the changed parameters are used to decide if the program should run under the debugger's control.

Related tasks

"Inactivating debugging profiles with the 3270 interface" on page 42

"Setting the display device" on page 34

Related reference

"The List profiles page" on page 17

Inactivating debugging profiles with the Web interface

About this task

To inactivate debugging profiles, start with the "List profiles" page, and follow these steps:

Procedure

1. Use the check boxes at the top of the page to ensure that the display includes the profiles you want to inactivate.
2. Scroll the list to a profile that you want to inactivate.
3. Select the profile using the check box to the left of the profile name.
4. Repeat steps 2 through 3 to select all the profiles you want to inactivate.
5. Click the **Inactivate** button.

Results

The "List profiles" page is refreshed.

Related reference

"The List profiles page" on page 17

Viewing a debugging profile with the Web interface

About this task

If you are not the owner of a debugging profile, you can view its contents, but you cannot change it. To view the contents of a debugging profile, start with the "List all profiles" page, and follow these steps:

Procedure

1. Use the check boxes at the top of the page to ensure that the profile you want to change is displayed.
2. Scroll the list to the profile you want to change.
3. Click on the profile name. CICS displays the "View compiled profile" page, or the "View Java profile" page.
4. Click on **List all profiles** to return to the "List all profiles" page.

Results

If you follow these steps for a profile that you own, CICS displays the "Edit compiled profile" page, or the "Edit Java profile" page. In this case, you will be able to modify the contents of the profile.

Related reference

"The List profiles page" on page 17

Changing a debugging profile with the Web interface

About this task

If you are the owner of a debugging profile, you can change its contents. Starting with the "List all profiles" page, follow these steps:

Procedure

1. Use the check boxes at the top of the page to ensure that the profile you want to change is displayed.
2. Scroll the list to the profile you want to change.
3. Click on the profile name. CICS displays the "Create compiled profile" page, or the "Create Java profile" page.
4. Make your changes to the displayed fields.
5. Click the **Replace** button.
6. Click on **List all profiles** to return to the "List all profiles" page.

Results

Note:

1. If you change the name of the profile, the debugging profile manager creates a new profile with the new name, and leaves the original profile unchanged.
2. If you follow these steps for a profile that you do not own, CICS displays the "View compiled profile" page, or the "View Java profile" page. In this case, you will not be able to modify the contents of the profile.
3. If you change a profile while it is active, the changes take effect immediately: the next time a program is started, the changed parameters are used to decide if the program should run under the debugger's control.

Related reference

"The List profiles page" on page 17

Copying debugging profiles with the Web interface

About this task

You can copy profiles that are owned by other users to create identical profiles that you own. You cannot copy a profile that you own. Each new profile has the same name as the one that was copied. Starting with the "List profiles" page, follow these steps:

Procedure

1. Use the check boxes at the top of the page to ensure that the display includes the profiles you want to copy.
2. Scroll the list to a profile that you want to copy.
3. Select the profile using the check box to the left of the profile name.
4. Repeat steps 2 through 3 to select all the profiles you want to copy.
5. Click the **Copy** button.

Results

The profiles are copied and the "List profiles" page is refreshed.

What to do next

If you want to create a new profile based on a profile that you own, follow the steps described in "Changing a debugging profile with the Web interface" on page 24.

Related reference

"The List profiles page" on page 17

Deleting debugging profiles with the Web interface

About this task

To delete debugging profiles, start with the "List profiles" page, and follow the steps below. You cannot delete a profile that is owned by another user.

Procedure

1. Use the check boxes at the top of the page to ensure that the display includes the profiles you want to delete.
2. Scroll the list to a profile you want to delete.
3. Select the profile using the check box to the left of the profile name.
4. Repeat steps 2 through 3 to select all the profiles you want to delete.
5. Click the **Delete** button.

Results

The profiles are deleted, and the "List profiles" page is refreshed.

Related reference

"The List profiles page" on page 17

Deleting the sample profiles

Although, in general, you cannot delete debugging profiles that are owned by another user, the sample profiles are handled as a special case, and you can delete them.

About this task

Be aware that, if you do delete the sample profiles:

- You may affect users who want to use the sample profiles.
- The only way to create them again is to have your system programmer re-initialize the debugging profiles data sets. However, if you do this, any other profiles which already exist will be deleted.

If you want to use the sample profiles, and you are concerned that other users may delete them, copy the samples. You will own the copies, and no-one else will be able to delete them.

If you don't want to see the sample profiles when you list profiles, display only the profiles that you own.

The Create compiled profile page

Use the "Create compiled profile" page to work with the contents of a debugging profile for compiled language (Language Environment) programs. You can use the page to perform the following functions:

Create a new profile
Initially, the page contains default values for some of the fields. You must supply other values to complete the profile.

Edit an existing profile
Initially, the page contains the values that were previously defined for the profile.

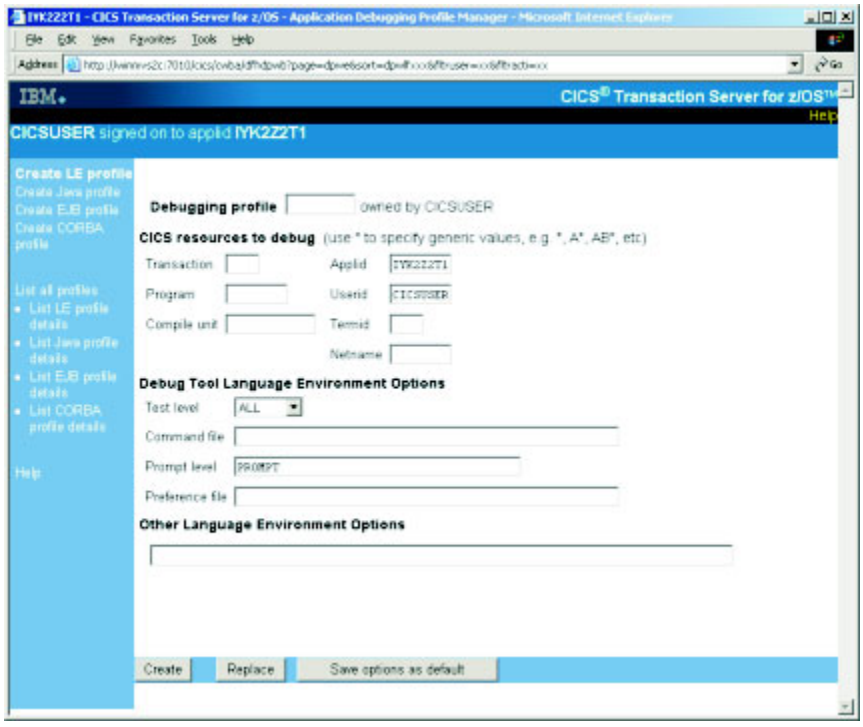


Figure 2. The Create compiled profile page

Related reference

"The List profiles page" on page 17

Fields on the Create compiled profile page

The fields on the "Create compiled profile" page are:

`Debugging profile`
Specifies the name of the profile.

If you are working with an existing profile, and you change the name that is displayed, the application debugging profile manager creates a new profile with the new name, and leaves the original profile unchanged.

Acceptable characters

A-Z 0-9 $ @ #

Any lower case characters you enter are converted to upper case.

The following fields specify which programs should trigger the start of a debugging session when the profile is active.

Transaction

Specify a value in this field when you want to debug only those programs that run under the specified transaction id.

Acceptable characters

A-Z a-z 0-9 $ @ # . / - _ % & ¢ ? ! : | " = ¬ , ; < >

Because transaction ids are case-sensitive, any lower case characters you enter are **not** converted to upper case.

You can specify a generic value if you want to debug programs that run under a set of similarly-named transactions.

Note: Do not specify an alias transaction name in this field; CICS does not support the use of alias transaction names to select programs for debugging.

Program

Specify a value in this field when you want to debug only the specified program. In this context the program is the program as known to CICS, such as a load module name, initial program in a transaction or a program that has been XCTL'd or LINKed to.

Acceptable characters

A-Z 0-9 $ @ #

Any lower case characters you enter are converted to upper case.

You can specify a generic value if you want to debug a set of similarly-named programs.

Compile unit

Specify a value in this field when you want to debug only the specified compile unit. You can specify a generic value if you want to debug a set of similarly-named compile units. In this context the compile unit is the program as known to the compiler; for example, PROGRAM-ID for COBOL and the main PROCEDURE name for PL/I.

Acceptable characters

A-Z a-z 0-9 $ @ # . / - _ % & ¢ ? ! : | " = ¬ , ; < >

Because compile unit names are case-sensitive, any lower case characters you enter are **not** converted to upper case.

Applid

Specify a value in this field when you want to confine debugging to programs that run in the specified CICS region.

Acceptable characters
A-Z 0-9 $ @ #
Any lower case characters you enter are converted to upper case.

You can specify a generic value if you want to debug programs in a set of similarly-named regions.

The default value is the applid of the region where the application debugging profile manager is running, and is displayed at the top of the page.

Userid
Specify a value in this field when you want to confine debugging to programs that are being run by the specified user. The default value is the ID of the user that is using the debugging profile manager.

Acceptable characters
A-Z 0-9 $ @ #
Any lower case characters you enter are converted to upper case.

You can specify a generic value if you want to debug programs that are being run by a group of similarly-named users.

The default value is the ID of the user that is using the application debugging profile manager, and is displayed at the top of the page.

Important: The user ID specified here is not necessarily the owner of the profile: the owner of the profile is the user who created it.

Termid
Specify a value in this field when you want to confine debugging to programs that are being run at the specified terminal.

Acceptable characters
A-Z a-z 0-9 $ @ # . / - _ % & ¢ ? ! : | " = ¬ , ; < >
Restriction: You cannot specify a terminal ID that consists entirely of blanks

You can specify a generic value if you want to debug programs that are being run at a number of similarly-named terminals.

Important: The terminal specified here is not necessarily the terminal at which the debugging session is conducted. The terminal used for the debugging session is specified in the “Set compiled display device” page.

Netname
Specify a value in this field when you want to confine debugging to programs that are being run at terminals with the specified netname.

Acceptable characters
A-Z 0-9 $ @ #
Any lower case characters you enter are converted to upper case.

You can specify a generic value if you want to debug programs that are being run at a number of terminals with similar netnames.

The following fields specify options that are passed to Debug Tool. See *Debug Tool for z/OS and OS/390 User's Guide* for more information. You can save the values that you specify; the saved values are used by default each time you create a Language Environment debugging profile.

Test level
Specifies what conditions need to be met for Debug Tool to gain control for programs that match this profile. Select one of the following values:
ALL
ERROR
NONE

Command file
Specifies the primary commands file associated with the profile. You can specify the fully qualified name of a sequential data set or a member of a partitioned data set.

Prompt level
Specifies whether an initial commands list is unconditionally executed during program initialization. Enter one of the following:
PROMPT
NOPROMPT
command

Preference file
Specifies the preference file that Debug Tool uses when debugging programs that match this profile. You can specify the fully qualified name of a sequential data set or a member of a partitioned data set.

Language Environment options
Specifies Language Environment runtime options for programs that match this profile. When a program is selected for debugging because it matches the profile, the runtime options specified will override other runtime options that you may have in effect. For more information about defining runtime options for Language Environment, see the *CICS Application Programming Guide*.

Buttons on the Create compiled profile page

The buttons on the "Create compiled profile" page are:

Create
Create a new profile using the information entered on the page

Replace
Update an existing profile using the information entered on the page

Save options as default
Save the contents of the following fields. The saved values are used by default each time you create a Language Environment debugging profile.
Test level
Command file
Prompt
Preference file
Language Environment options

The Create Java profile page

Use the "Create Java profile" page to work with the contents of a debugging profile for Java programs. You can use the page to perform the following functions:

Create a new profile
Initially, the page contains default values for some of the fields. You must supply other values to complete the profile.

Edit an existing profile
Initially, the page contains the values that were previously defined for the profile.

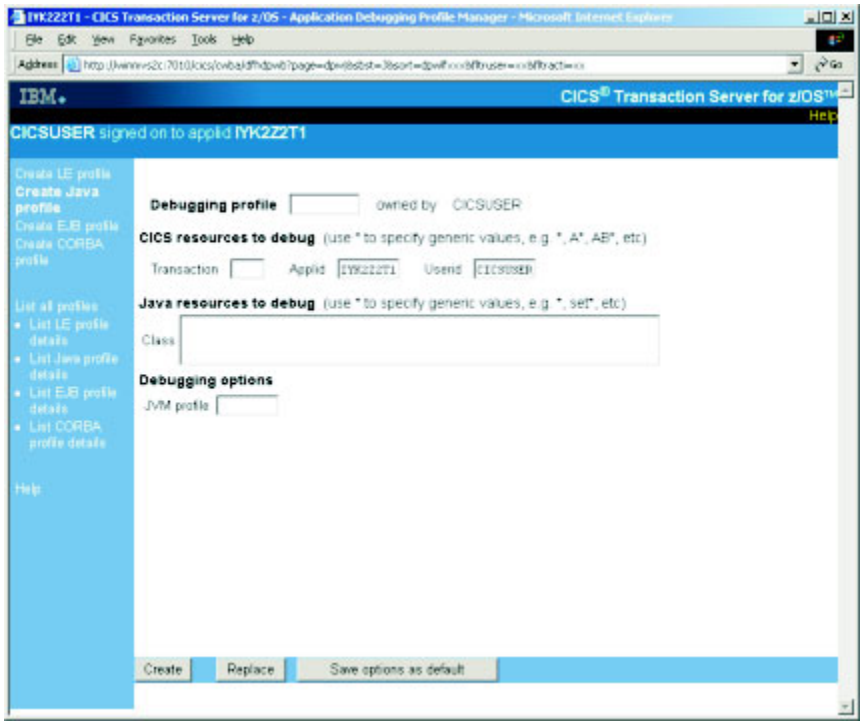


Figure 3. The Create Java profile page

Related reference
"The List profiles page" on page 17

Fields on the Create Java profile page

The fields on the "Create Java profile" page are:

`Debugging Profile`
Specifies the name of the profile.

If you are working with an existing profile, and you change the name that is displayed, the application debugging profile manager creates a new profile with the new name, and leaves the original profile unchanged.

Acceptable characters
`A-Z 0-9 $ @ #`
Any lower case characters you enter are converted to upper case.

The following fields specify which programs should trigger the start of a debugging session when the profile is active:

Transaction
Specify a value in this field when you want to debug only those programs that run under the specified transaction id.

Acceptable characters
A-Z a-z 0-9 $ @ # . / - _ % & ¢ ? ! : | " = ¬ , ; < >

You can specify a generic value if you want to debug programs that run under a set of similarly name transactions.

Note: Do not specify an alias transaction name in this field; CICS does not support the use of alias transaction names to select programs for debugging.

Applid
Specify a value in this field when you want to confine debugging to programs that run in the specified CICS region. The default value is the applid of the region where the Debugging Profile Manager is running.

Acceptable characters
A-Z 0-9 $ @ # Any lower case characters you enter are converted to upper case.

You can specify a generic value if you want to debug programs in a set of similarly-named regions.

Userid
Specify a value in this field when you want to confine debugging to programs that are being run by the specified user. The default value is the ID of the user that is using the debugging profile manager.

Acceptable characters
A-Z 0-9 $ @ # Any lower case characters you enter are converted to upper case.

You can specify a generic value if you want to debug programs that are being run by a group of similarly-named users.

Important: The user ID specified here is not necessarily the owner of the profile: the owner of the profile is the user who created it.

The following fields specify which Java resources should trigger the start of a debugging session when the profile is active:

Type
Specifies the type of Java resource that you want to debug:
Java Select this value when you want to debug a Java program.
EJB Select this value when you want to debug an enterprise bean.
CORBA
Select this value when you want to debug a stateless CORBA object.

Class
For Java programs and stateless CORBA objects only, specify a value in this field when you want to debug only the specified class. You can specify a generic value if you want to debug a set of similarly-named classes.

Bean
For enterprise beans only, specify a value in this field when you want to debug only the specified bean. You can specify a generic value if you want to debug a set of similarly-named beans.

Method
For enterprise beans and stateless CORBA objects only, specify a value in this field when you want to debug only the specified method.

When an inbound request initiated by a Java remote method invocation is received, the value specified is compared with the mangled name in the inbound request to determine if the profile matches the request. If it is possible that mangling can take place, do not specify a method name in the debugging profile, but specify a generic method instead. For more information about mangling of method names, see *Java Applications in CICS*.

The following field specifies the debugging options for this profile. You can save the value that you specify; the saved value is used by default each time you create a Java debugging profile.

JVM profile
Specifies the name of the JVM profile that is used for Java programs that match this profile. The profile should specify that the Java program is to run in debug mode. You cannot specify a generic value for this parameter.

Buttons on the Create Java profile page

The buttons on the "Create Java profile" page are:

Create
Create a new profile using the information entered on the page

Replace
Update an existing profile using the information entered on the page

Save options as default
Save the contents of the following field. The saved value is used by default each time you create a Java debugging profile.
JVM profile

The View LE profile page

Use the "View LE profile" page to view the contents of a debugging profile for compiled language programs. You cannot change the profile with this page. The information displayed is described in "The Create compiled profile page" on page 26.

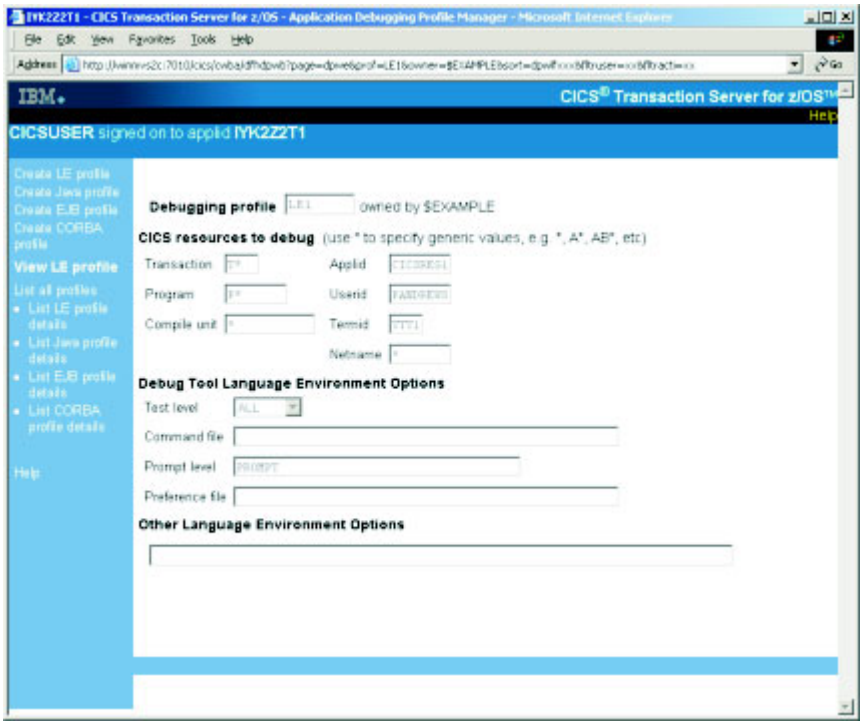


Figure 4. The View LE profile page

Related reference

"The List profiles page" on page 17

The View Java profile page

Use the "View Java profile" page to view the contents of a debugging profile for compiled language programs. You cannot change the profile with this page. The information displayed is described in "The Create Java profile page" on page 29.

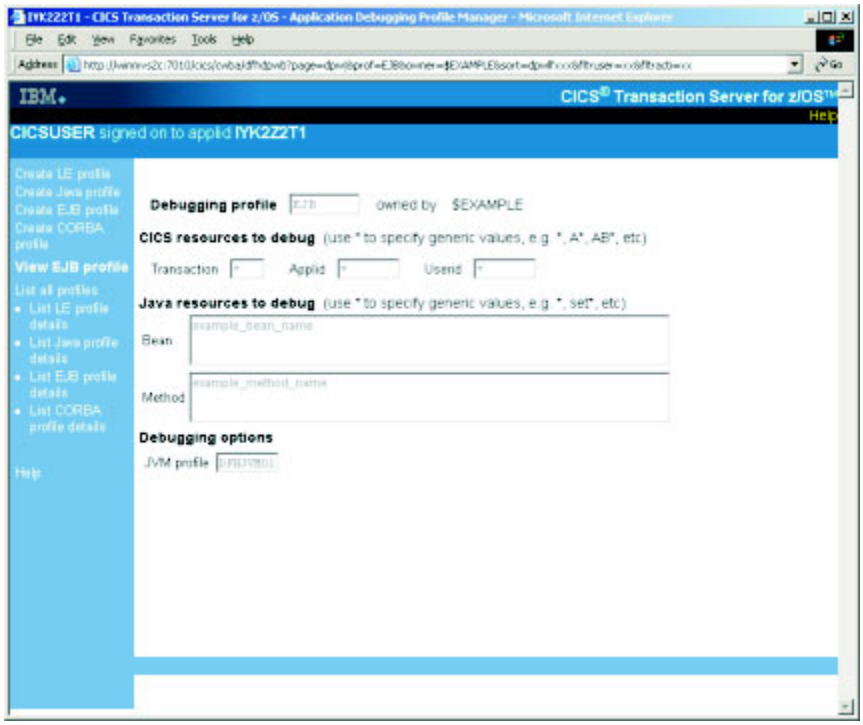


Figure 5. The View Java profile page

Related reference

"The List profiles page" on page 17

Setting the display device

About this task

When you have created a debugging profile for a compiled language (Language Environment) program, but before you can start debugging the application programs defined in the profile, you must specify the display device with which you will interact with the debugger. You can use:

a 3270 terminal
a debugging tool on a workstation.

The display device becomes associated with a debugging profile when you activate the profile, and remains associated with the profile until you make the profile inactive.

You can choose when you specify the display device:

- You can specify the display device *before* you make a profile active. The same display device will be associated with each profile that you subsequently make active.
- You can specify the display device *when* you make a profile active. The same display device will be associated with the profile that is made active. If you make more than one profile active at the same time (by selecting a number of profiles on the "List debugging profile" page) the same display device will be associated with them all.

Related tasks

"Activating debugging profiles with the Web interface" on page 22

Related reference

"The List profiles page" on page 17

Specifying the display device before you make a profile active

Starting with the "List profiles" page, follow these steps:

1. Click on Set LE display device. CICS displays the "Set LE display device" page.
2. Complete the details of the display device that you want to associate with the profile.
3. Select "In the future, do not show this page when activating profiles".
4. Click the **Save and return** button. CICS saves the display device settings, and displays the "List profiles" page.

The "Set LE display device" page will not be displayed when you activate profiles; the settings you have supplied will be applied to all profiles for compiled language programs when you activate them.

Specifying the display device when you make a profile active

About this task

Starting with the "List profiles" page, follow these steps:

Procedure

1. Click on Set LE display device. CICS displays the "Set LE display device" page.
2. Complete the details of the display device that you want to associate with the profile.
3. Using the check box, deselect "In the future, do not show this page when activating profiles".
4. Click the **Save and return** button. CICS saves the display device settings, and displays the "List profiles" page.

Results

CICS will display the "Set LE display device" page whenever you activate profiles for compiled language profiles. The page is initialized with the last settings you supplied. To change the settings:

1. Change any details of the display device that you want to associate with the set of profiles that is being activated.
2. Click the **Save and return** button. CICS saves the display device settings, and displays the "List profiles" page.

The Set compiled display device page

Use the "Set compiled display device" page to specify the display device with which you will interact with the debugger.

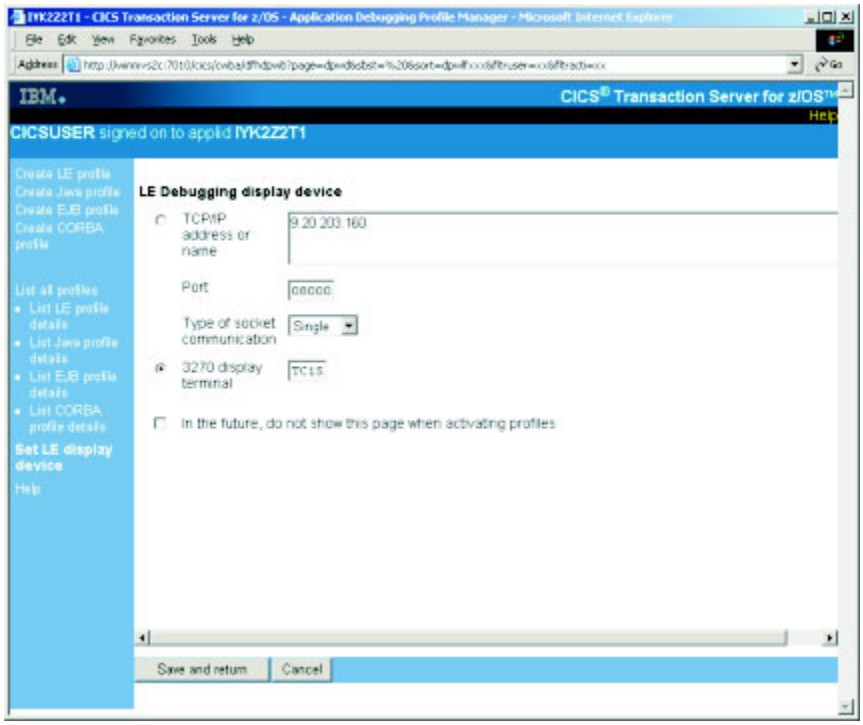


Figure 6. The Set LE display device page

Related reference

"The List profiles page" on page 17

Fields on the Set compiled display device page

The fields on the "Set compiled display device" page are:

Debugging display device
Use the radio buttons to select how you will interact with the debugger:

TCP/IP address or name
Specifies that you will interact with the debugger using a debugging client on your workstation. Supply the following information:

- The IP address or name of the host where the debugging client is running. By default, CICS inserts the IP address of the client which is running the browser, or — if there is a firewall between the browser and CICS — the IP address of the firewall.
- The port number at which the debugging client listens for a connection. Specify a value in the range 0 - 65535. The default is 8000.

Type of socket communication
For a debugging client on your workstation, specifies whether the debugging client and debugging server will communicate using a single socket or more than one socket.

Single
Use a single socket for communication. This is the default value, and is the preferred value when you use a WebSphere® Studio product as your debugging client.

Multiple
Use more than one socket for communication. You must specify this value when you use a VisualAge® product as your debugging client.

3270 display terminal
Specifies that you will interact with the debugger using a 3270 type terminal. Supply the following information:

- The terminal id of the terminal at which you will interact with the debugger.

  Important: The terminal specified here is not necessarily the terminal at which the transaction being debugged will run.

Buttons on the Set compiled display device page

The buttons on the "Set compiled display device" page are:

Save and return
Save the settings and return to the "List all profiles" page

Cancel
Return to the "List all profiles" page without saving the settings

Using the application debugging profile manager 3270 interface

The CADP transaction displays debugging profile information by means of the application debugging profile manager 3270 interface.

About this task

If you are debugging TCP/IP information, note that the port number must be specified in the PORT field of your resource. If the port is specified as part of the HOST attribute, the port information is not available to the CADP transaction.

Procedure

1. On a 3270 screen, enter `CADP`.
2. Press `ENTER`. CICS displays the "List debugging profiles" screen.
3. Select your next action from the list of subtopics.

The List debugging profiles screen

Use the "List debugging profiles" screen to display a list of debugging profiles. When you start the debugging profile manager, CICS displays the "List debugging profiles" screen.

```
CADP      -    CICS Application Debugging Profile Manager   -     IYK2Z2T1

List Debugging Profiles    (A=Activate,I=Inactivate,D=Delete,C=Copy)

  Owner    Profile  S Tran Program  Compile Unit Applid   Userid   Term Type
_ $EXAMPLE CORBA    I T*                         *        IORWERTH      Corb
_ CICSUSER CORBA    I T*                         *        IORWERTH      Corb
_ $EXAMPLE EJB      I *                          *        *             EJB
_ $EXAMPLE JAVA     I TR*                        *        PENFOLD*      Java
_ $EXAMPLE LE1      I T*   P*       *            CICSREG1 PANDREWS TTT1 LE
_ $EXAMPLE LE2      I TR   *        SAMPCOMPUN + CICSREG2 DRBEARD* TTT2 LE
_ $EXAMPLE LE3      I TRN3 PROG3    *            CICSREG3 *        TTT2 LE

7   profile(s). All profiles shown
Enter=Process PF1=Help 2=Filter 3=Exit 4=View 5=Create LE 6=Create Java
                9=Set display device 10=Edit 11=Sort
```

Figure 7. The List debugging profiles screen, showing the example profiles

When you use the debugging profile manager for the first time, CICS displays all the profiles that you own. Subsequently, CICS displays the profiles that were selected when you last used it.

If there are more profiles than can be displayed on one screen, use PF7 and PF8 to scroll backwards and forwards through the list. If you have no profiles, CICS displays an empty list.

The list contains selected information from the debugging profiles. The columns on the screen are:

Owner
The userid of the profile owner; that is, of the user who created the profile.

Profile
The name of the profile

S The status of the profile (**A** for Active, or **I** for Inactive)

The following columns display information specified when the profile is created:

Tran Displays the contents of the **transaction** field

Program
Displays the contents of the **program** field

Compile Unit
Displays the the first ten characters of the **Compile Unit** field. If the Compile Unit name is longer, a + to the right of the name shows that only part of the name is displayed.

Applid
Displays the contents of the **Applid** field

Userid
Displays the contents of the **Userid** field

Term Displays the contents of the **Terminal** field

Type Displays the type of program specified in the debugging profile:

Corb CORBA object
EJB Enterprise bean
Java Java program
LE Compiled language program

Not all the information in the debugging profile is displayed on the "List debugging profiles" screen. To display the additional information, move the cursor to the line that contains the profile, and press PF4.

You can change the way CICS displays information on the "List debugging profiles" screen:

Selecting which profiles are displayed

Use PF2 to cycle through the available options in turn. The options are

1. Display all the profiles in the system. This is the setting the first time you use the debugging profile manager
2. Display the profiles that you own
3. Display all active profiles

Sorting the list

Use PF11 to cycle through the available options in turn. The options are:

1. Re-display the profiles in sequence of profile name. This is the sequence the first time you use the debugging profile manager
2. Re-display the profiles in sequence of transaction ID
3. Re-display the profiles in sequence of program name
4. Re-display the profiles in sequence of owner

In each case, CICS uses the EBCDIC sorting sequence.

Your choice of which profiles are displayed, and your chosen sequence, are preserved, and used the next time you use the debugging profile manager.

Related tasks

"Creating a debugging profile with the 3270 interface" on page 40
"Activating debugging profiles with the 3270 interface" on page 41
"Inactivating debugging profiles with the 3270 interface" on page 42
"Viewing a debugging profile with the 3270 interface" on page 42
"Changing a debugging profile with the 3270 interface" on page 42
"Copying debugging profiles with the 3270 interface" on page 43
"Deleting debugging profiles with the 3270 interface" on page 44
"Setting the display device" on page 55

Related reference

"The Create Java debugging profile screen" on page 49
"The Create LE debugging profile screen" on page 45
"The View Java debugging profile screen" on page 53
"The View LE debugging profile screen" on page 52
"The Set LE debugging display device screen" on page 56

Function keys for the List debugging profiles screen

The function keys for the "List debugging profiles" screen are:

PF1
Display the help screen

PF2
Selects which debugging profiles are displayed. This key cycles through the available options in turn.

PF3
End the debugging profile manager
PF4
Displays the “View LE debugging profile” screen or the “View Java debugging profile” screen for the profile on the line that contains the cursor.
PF5
Create a new debugging profile for a compiled language program
PF6
Create a new debugging profile for a Java program
PF7
Scroll backwards
PF8
Scroll forwards
PF9
Display the “Set LE debugging display device” screen
PF10
Edit the profile on the line that contains the cursor, using the “Create LE debugging profile” screen or the “Create Java debugging profile” screen.
PF11
Re-display the debugging profiles in a different sequence. This key cycles through the available options in turn.

Creating a debugging profile with the 3270 interface

About this task

You can create debugging profiles in these ways:

Procedure

- You can create a completely new profile by entering all the information needed to define the profile.
- You can base the new profile on an existing profile.

Related reference

“The List debugging profiles screen” on page 37

“The Create LE debugging profile screen” on page 45

“The Create Java debugging profile screen” on page 49

Creating a new profile

About this task

Starting with the “List debugging profiles” screen, follow these steps:

Procedure

1. Decide which type of profile you want to create.
   - To create a profile for a compiled language program, press PF5. CICS displays the “Create LE debugging profile” screen.
   - To create a profile for a Java program, press PF6. CICS displays the “Create Java debugging profile” screen.
2. Complete the fields that you need to specify your profile. You might need to use PF7 and PF8 to scroll the display.
3. Press ENTER. CICS checks that you have entered valid data.
   - If you have specified valid data, the profile is saved

- If your data contains an error, CICS displays a message. Re-enter the data, and press ENTER again.

4. Press PF12 to return to the "List debugging profiles" screen.

Basing a new profile on an existing profile

About this task

You can create a new debugging profile using an existing profile as a starting point. The steps you take depend upon whether the original profile is owned by you or another user:

If you own the profile
Follow the steps described in "Changing a debugging profile with the 3270 interface" on page 42. Before you save the profile, give it a new name.

If another user owns the profile
Follow the steps below.

Procedure

1. Copy the profile, following the steps described in "Copying debugging profiles with the 3270 interface" on page 43.
2. Make any changes to the profile by following the steps described in "Changing a debugging profile with the 3270 interface" on page 42.

Activating debugging profiles with the 3270 interface

About this task

To activate debugging profiles, start with the "List debugging profiles" screen, and follow these steps:

Procedure

1. Use PF2 to ensure that the display includes the profiles you want to activate.
2. Use PF7 and PF8 to scroll to a profile that you want to activate.
3. Type **A** (for Activate) in the field to the left of the profile name.
4. Repeat steps 2 through 3 to select all the profiles you want to activate.
5. Press ENTER. By default, if any of the selected profiles is for a compiled language program, CICS displays the "Set LE debugging display device" screen.

   If none of the selected profiles is for a compiled language program, CICS refreshes the "List debugging profiles" screen.

   You can choose not to see the "Set LE debugging display device" screen when you activate profiles. See "Setting the display device" on page 55 for more information.

Results

Note: If you change a profile while it is active, the changes take effect immediately: the next time a program is started, the changed parameters are used to decide if the program should run under the debugger's control.

Related reference

"The List debugging profiles screen" on page 37

"The Set LE debugging display device screen" on page 56

Inactivating debugging profiles with the 3270 interface

About this task

To inactivate debugging profiles, start with the "List debugging profiles" screen, and follow these steps:

Procedure

1. Use PF2 to ensure that the display includes the profiles you want to inactivate.
2. Use PF7 and PF8 to scroll to a profile that you want to inactivate.
3. Type **I** (for Inactivate) in the field to the left of the profile name.
4. Repeat steps 2 through 3 to select all the profiles you want to inactivate.
5. Press ENTER. CICS makes the selected profiles inactive, and refreshes the "List debugging profiles" screen.

Related tasks

"Activating debugging profiles with the Web interface" on page 22

Related reference

"The List debugging profiles screen" on page 37

Viewing a debugging profile with the 3270 interface

About this task

To view the contents of a debugging profile, without changing it, start with the "List debugging profiles" screen, and follow these steps:

Procedure

1. Use PF2 to ensure that the display includes the profile you want to view.
2. Use PF7 and PF8 to scroll to the profile you want to view.
3. Move the cursor to the line containing the profile that you want to view.
4. Press PF4. CICS displays the "View LE debugging profile" screen, or the "View Java debugging profile", depending on the options that were selected when the debugging profile was created.
5. When you have finished viewing the profile, press PF12 to return to the "List debugging profiles" screen.

Related reference

"The List debugging profiles screen" on page 37

"The View Java debugging profile screen" on page 53

"The View LE debugging profile screen" on page 52

Changing a debugging profile with the 3270 interface

About this task

If you are the owner of a debugging profile, you can change its contents. Starting with the "List debugging profiles" screen, follow these steps:

Procedure

1. Use PF2 to ensure that the display includes the profile you want to change.
2. Use PF7 and PF8 to scroll to a profile you want to change.
3. Move the cursor to the line containing the profile that you want to change.
4. Press PF10. CICS displays “Create LE debugging profile” screen, or the “Create Java debugging profile” screen, depending on the options that were selected when the debugging profile was created.
5. Make your changes to the fields displayed on the screen.
6. Press PF10. CICS saves the changed profile.
7. Press PF12 to return to the “List debugging profiles” screen.

Results

Note:

1. If you change the name of the profile, the debugging profile manager creates a new profile with the new name, and leaves the original profile unchanged.
2. If you change a profile while it is active, the changes take effect immediately: the next time a program is started, the changed parameters are used to decide if the program should run under the debugger's control.

Related reference

“The List debugging profiles screen” on page 37

“The Create LE debugging profile screen” on page 45

“The Create Java debugging profile screen” on page 49

Copying debugging profiles with the 3270 interface

About this task

You can copy profiles that are owned by other users, to create identical profiles that you own. Each new profile has the same name as the one that was copied. Starting with the “List debugging profiles” screen, follow these steps:

Procedure

1. Use PF2 to ensure that the display includes the profiles you want to copy.
2. Use PF7 and PF8 to scroll to a profile that you want to copy.
3. Type **C** (for Copy) in the field to the left of the profile name.
4. Repeat steps 2 through 3 to select all the profiles you want to copy.
5. Press ENTER. CICS copies the profiles, and refreshes the “List debugging profiles” screen.

Results

Note: You cannot copy a profile that you own. If you want to create a new profile based on a profile that you own, follow the steps described in “Changing a debugging profile with the 3270 interface” on page 42. Before saving the changed profile, give it a new name.

Related tasks

"Combining actions on the List debugging profiles screen"

Related reference

"The List debugging profiles screen" on page 37

Deleting debugging profiles with the 3270 interface

About this task

To delete debugging profiles, start with the "List debugging profiles" screen, and follow the steps below. You cannot delete a profile that is owned by another user.

Procedure

1. Use PF2 to ensure that the display includes the profiles you want to delete.
2. Use PF7 and PF8 to scroll to a profile you want to delete.
3. You must make the profile inactive before you can delete it. To do this, see "Inactivating debugging profiles with the 3270 interface" on page 42.
4. Type **D** (for Delete) in the field to the left of the profile name.
5. Repeat steps 2 through 4 to select all the profiles you want to delete.
6. Press ENTER. CICS deletes the selected profiles, and refreshes the "List debugging profiles" screen.

Related reference

"The List debugging profiles screen" on page 37

Deleting the sample profiles

About this task

Although, in general, you cannot delete debugging profiles that are owned by another user, the sample profiles are handled as a special case, and you can delete them. Be aware that, if you do delete the sample profiles:

- You may affect users who want to use the sample profiles.
- The only way to create them again is to have your system programmer re-initialize the debugging profiles data sets. However, if you do this, any other profiles which already exist will be deleted.

If you want to use the sample profiles, and you are concerned that other users may delete them, copy the samples. You will own the copies, and no-one else will be able to delete them.

If you don't want to see the sample profiles when you list profiles, use PF2 on the "List debugging profiles" screen to display only the profiles that you own.

Combining actions on the List debugging profiles screen

About this task

From the "List debugging profiles screen", you can activate, inactivate, delete and copy debugging profiles by typing the appropriate action character (**A**, **I**, **D** and **C** respectively) in the field to the left of the profile name. You can combine these actions on the "List debugging profiles" screen. For example, you can activate some profiles, and inactivate others in a single operation:

Procedure

1. Use PF2 to ensure that the display includes the profiles you want to work with.
2. Use PF7 and PF8 to scroll to a profile you want to work with.
3. Type the action character in the field to the left of the profile name.
4. Repeat steps 2 and 3 to select all the profiles you want to work with.
5. Press ENTER. CICS processes the selected profiles, and refreshes the "List debugging profiles" screen. If any of the selected actions fail, processing stops, and the unprocessed action characters remain on the screen.

Related tasks

"Copying debugging profiles with the 3270 interface" on page 43

Related reference

"The List profiles page" on page 17

The Create LE debugging profile screen

Use the "Create LE debugging profile" screen to work with the contents of a debugging profile for compiled language programs. You can use the screen to perform the following functions:

Create a new profile
Initially, the screen contains default values for some of the fields. You must supply other values to complete the profile.

Edit an existing profile
Initially, the screen contains the values that were previously defined for the profile.

```
CADP      -    CICS Application Debugging Profile Manager   -     IYK2Z2T1

Create LE Debugging Profile ==>           for CICSUSER


CICS Resources To Debug (use * to specify generic values e.g. *, A*, AB*, etc.)
Transaction      ==>                                  Applid   ==> IYK2Z2T1
Program          ==>                                  Userid   ==> CICSUSER
Compile Unit     ==>                                  Termid   ==> TC15
                                                      Netname  ==> IYCWTC15

Debug Tool Language Environment Options
Test Level       ==> All                                      (All,Error,None)
Command File     ==>
Prompt Level     ==> PROMPT
Preference File  ==>

Other Language Environment Options
==>
==>
==>
==>

Enter=Create PF1=Help 2=Save options as defaults 3=Exit 10=Replace 12=Return
```

Figure 8. The CADP Create LE debugging profile screen

Related tasks

"Creating a debugging profile with the 3270 interface" on page 40

"Changing a debugging profile with the 3270 interface" on page 42

Related reference

"The List debugging profiles screen" on page 37

Fields on the Create LE debugging profile screen

The fields on the "Create LE debugging profile" screen are:

Create LE Debugging Profile

Specify the name of the profile. If you change the name of an existing profile, CADP creates a new profile with the new name, and leaves the original profile unchanged.

Acceptable characters
A-Z 0-9 $ @ #
Any lower case characters you enter are converted to upper case.

The following fields specify which programs should trigger the start of a debugging session when the profile is active.

Transaction

Specify a value in this field when you want to debug only those programs that run under the specified transaction id.

Acceptable characters
A-Z a-z 0-9 $ @ # . / - _ % & ¢ ? ! : | " = ¬ , ; < >
Because transaction ids are case-sensitive, any lower case characters you enter are **not** converted to upper case.

You can specify a generic value if you want to debug programs that run under a set of similarly-named transactions.

Note: Do not specify an alias transaction name in this field; CICS does not support the use of alias transaction names to select programs for debugging.

Program

Specify a value in this field when you want to debug only the specified program. In this context the program is the program as known to CICS, such as a load module name, initial program in a transaction or a program that has been XCTL'd or LINKed to.

Acceptable characters
A-Z 0-9 $ @ #
Any lower case characters you enter are converted to upper case.

You can specify a generic value if you want to debug a set of similarly-named programs.

Compile unit

Specify a value in this field when you want to debug only the specified compile unit. You can specify a generic value if you want to debug a set of similarly-named compile units. In this context the compile unit is the program as known to the compiler; for example, PROGRAM-ID for COBOL and the main PROCEDURE name for PL/I.

Acceptable characters

A-Z a-z 0-9 $ @ # . / - _ % & ¢ ? ! : | " = ¬ , ; < >

Because compile unit names are case-sensitive, any lower case characters you enter are **not** converted to upper case.

Applid

Specify a value in this field when you want to confine debugging to programs that run in the specified CICS region. The default value is the applid of the region where the Debugging Profile Manager is running.

Acceptable characters

A-Z 0-9 $ @ #

Any lower case characters you enter are converted to upper case.

You can specify a generic value if you want to debug programs in a set of similarly-named regions.

Userid

Specify a value in this field when you want to confine debugging to programs that are being run by the specified user. The default value is the ID of the user who is using the debugging profile manager.

Acceptable characters

A-Z 0-9 $ @ #

Any lower case characters you enter are converted to upper case.

You can specify a generic value if you want to debug programs that are being run by a group of similarly-named users.

Important: The user ID specified here is not necessarily the owner of the profile: the owner of the profile is the user who created it.

Termid

Specify a value in this field when you want to confine debugging to programs that are being run at the specified terminal.

Acceptable characters

A-Z a-z 0-9 $ @ # . / - _ % & ¢ ? ! : | " = ¬ , ; < >

Restriction: You cannot specify a terminal ID that consists entirely of blanks

You can specify a generic value if you want to debug programs that are being run at a number of similarly-named terminals.

Important: The terminal specified here is not necessarily the terminal at which the debugging session is conducted. The terminal used for the debugging session is specified in the "Set LE debugging display device" screen.

Netname

Specify a value in this field when you want to confine debugging to programs that are being run at terminals with the specified netname.

Acceptable characters

A-Z 0-9 $ @ #

Any lower case characters you enter are converted to upper case.

You can specify a generic value if you want to debug programs that are being run at a number of terminals with similar netnames.

The following fields specify suboptions of the TEST runtime option, and are passed to Debug Tool. See the *Debug Tool for z/OS User's Guide* for more information. You can save the values that you specify; the saved values are used by default each time you create an LE debugging profile. For more information, see "Specifying default values for Debug Tool and LE options."

Test level
Specifies which conditions raised by your program will cause Debug Tool to gain control. You can enter the following values:
All
Error
None

Command file
Specifies the primary commands file associated with the profile. You can specify the fully qualified name of a sequential data set or a member of a partitioned data set.

Prompt level
Specifies whether an initial commands list is unconditionally executed during program initialization. Enter one of the following:
PROMPT
NOPROMPT
command

Preference file
Specifies the preference file that Debug Tool uses when debugging programs that match this profile. You can specify the fully qualified name of a sequential data set or a member of a partitioned data set.

Other Language Environment options
Specifies Language Environment runtime options for programs that match this profile. When a program is selected for debugging because it matches the profile, the runtime options specified will override other runtime options that you may have in effect. For more information, see the *CICS Application Programming Guide*.

Function keys for the Create LE debugging profile screen

The function keys for the "Create LE debugging profile" screen are:

PF1
Display the help screen

PF2
Save the contents of the Debug Tool options, and the Language Environment options. See "Specifying default values for Debug Tool and LE options."

PF3
End the debugging profile manager

PF10
Update an existing profile with the information on the screen

PF12
Return to the "List debugging profiles" screen

Specifying default values for Debug Tool and LE options: You can specify default values for the following Debug Tool options, and the Language Environment options. The saved values are used by default each time you create a debugging profile for a compiled language program. The Debug Tool options are:

Test level
Command file
Prompt level
Preference file

To save the default values, start with the "Create LE debugging profile" screen, and follow these steps:

1. Type the default values that you want to specify for the Debug Tool options and Language Environment options
2. Press PF2. CICS saves the values that you have specified.

The values that you save will be used by default each time you create a new profile.

The Create Java debugging profile screen

Use the "Create Java debugging profile" screen to work with the contents of a debugging profile for Java programs. You can use the screen to perform the following functions:

Create a new profile
Initially, the screen contains default values for some of the fields. You must supply other values to complete the profile.

Edit an existing profile
Initially, the screen contains the values that were previously defined for the profile.

The screen is in two parts; use PF7 and PF8 to scroll between them.

```
CADP      -    CICS Application Debugging Profile Manager   -     IYK2Z2T1

Create Java Debugging Profile ==>            for CICSUSER

CICS Resources To Debug (use * to specify generic values e.g. *, A*, AB*, etc.)
Transaction      ==>                          Applid       ==> IYK2Z2T1
                                              Userid       ==> CICSUSER

Debugging Options
JVM Profile      ==>

Java Resources To Debug
Type             ==> J       (J=Java Applications, E=Enterprise Beans, C=Corba)

Class (Java Applications or Corba)
==>
==>
==>
==>

Press PF8 to set Bean and Method

Enter=Create PF1=Help 2=Save options as defaults 3=Exit 8=Forward
10=Replace 12=Return
```

```
CADP      -    CICS Application Debugging Profile Manager   -     IYK2Z2T1

Java Resources To Debug

Bean (Enterprise Beans only)
==>
==>
==>
==>

Method (Enterprise Beans or Corba)
==>
==>
==>
==>









Enter=Create PF1=Help 3=Exit 7=Back 10=Replace 12=Return
```

Figure 9. The CADP Create Java debugging profile screens

Related tasks

"Creating a debugging profile with the 3270 interface" on page 40

"Changing a debugging profile with the 3270 interface" on page 42

Related reference

"The List debugging profiles screen" on page 37

Fields on the Create Java debugging profile screen

The fields on the "Create Java debugging profile" screen are:

Create Java Debugging Profile
Specifies the name of the profile. If you change the name of an existing profile, CADP creates a new profile with the new name, and leaves the original profile unchanged.

Acceptable characters
A-Z 0-9 $ @ #
Any lower case characters you enter are converted to upper case.

The following fields specify which programs should trigger the start of a debugging session when the profile is active.

Transaction
Specify a value in this field when you want to debug only those programs that run under the specified transaction id.

Acceptable characters
A-Z a-z 0-9 $ @ # . / - _ % & ¢ ? ! : | " = ¬ , ; < >

You can specify a generic value if you want to debug programs that run under a set of similarly-named transactions.

Note: Do not specify an alias transaction name in this field; CICS does not support the use of alias transaction names to select programs for debugging.

Applid
Specify a value in this field when you want to confine debugging to programs that run in the specified CICS region. The default value is the applid of the region where the Debugging Profile Manager is running.

Acceptable characters
A-Z 0-9 $ @ #
Any lower case characters you enter are converted to upper case.

You can specify a generic value if you want to debug programs in a set of similarly-named regions.

Userid
Specify a value in this field when you want to confine debugging to programs that are being run by the specified user. The default value is the ID of the user that is using the debugging profile manager.

Acceptable characters
A-Z 0-9 $ @ #
Any lower case characters you enter are converted to upper case.

You can specify a generic value if you want to debug programs that are being run by a group of similarly-named users.

Important: The user ID specified here is not necessarily the owner of the profile: the owner of the profile is the user who created it.

The following field specifies the debugging options for this profile. You can save the value that you specify; the saved value is used by default each time you create a Java debugging profile. To save the value, press PF2.

JVM profile
Specifies the name of the JVM profile that is used for Java programs that match this profile. The profile should specify that the Java program is to run in debug mode. For more information about the JVM profile, see Setting up JVM profiles and JVM properties files *Java Applications in CICS*.

The following fields specify which Java resources should trigger the start of a debugging session when the profile is active:

Type
Specifies the type of Java resource that you want to debug:
J Enter this value when you want to debug a Java program.
E Enter this value when you want to debug an enterprise bean.
C Enter this value when you want to debug a stateless CORBA object.

Class
For Java programs and stateless CORBA objects only, specify a value in this field when you want to debug only the specified class. You can specify a generic value if you want to debug a set of similarly-named classes.

Bean
For enterprise beans only, specify a value in this field when you want to debug only the specified bean. You can specify a generic value if you want to debug a set of similarly-named beans.

Method
For enterprise beans and CORBA objects only, specify a value in this field when you want to debug only the specified method.

When an inbound request initiated by a Java remote method invocation is received, the value specified is compared with the mangled name in the inbound request to determine if the profile matches the request. If it is possible that mangling can take place, do not specify a method name in the debugging profile, but specify a generic method instead. For more information about mangling of method names, see *Java Applications in CICS*.

Function keys for the Create Java debugging profile screen

The function keys for the "Create Java debugging profile" screen are:

PF1
Display the help screen

PF2
Save the contents of the **JVM profile** field. The saved value is used by default each time you create a Java debugging profile.

PF3
End the debugging profile manager

PF7
Scroll forward

PF8
Scroll backward

PF10
Update an existing profile with the information on the screen

PF12
Return to the "List debugging profiles" screen

The View LE debugging profile screen

Use the "View LE debugging profiles" screen to view the contents of a debugging profile for compiled language programs. You cannot change the profile with this screen.

```
CADP      -    CICS Application Debugging Profile Manager   -     IYK2Z2T1

View LE Debugging Profile LE1      for $EXAMPLE


CICS Resources To Debug
Transaction      ==> T*                              Applid   ==> CICSREG1
Program          ==> P*                              Userid   ==> PANDREWS
Comp Unit        ==> *                               Termid   ==> TTT1
                                                     Netname  ==> *

Debug Tool Language Environment Options
Test Level       ==> All                                    (All,Error,None)
Command File     ==>
Prompt Level     ==> PROMPT
Preference File  ==>

Other Language Environment Options
==>
==>
==>
==>

PF1=Help 3=Exit 12=Return
```

Figure 10. The View LE debugging profile screen, showing example profile LE1

The information displayed is described in "The Create LE debugging profile screen" on page 45.

Related tasks

"Viewing a debugging profile with the 3270 interface" on page 42

Related reference

"The List debugging profiles screen" on page 37

Function keys for the View LE debugging profile screen

The function keys for the "View LE debugging profile" screen are:

PF1
: Display the help screen

PF3
: End the debugging profile manager

PF12
: Return to the "List debugging profiles" screen

The View Java debugging profile screen

Use the "View Java debugging profiles" screen to view the contents of a debugging profile for Java programs. You cannot change the profile with this screen.

The screen is in two parts; use PF7 and PF8 to scroll between them.

```
CADP      -    CICS Application Debugging Profile Manager   -     IYK2Z2T1

View Java Debugging Profile EJB      for $EXAMPLE

CICS Resources To Debug
Transaction      ==> *                        Applid       ==> *
                                              Userid       ==> *

Debugging Options
JVM Profile      ==> DFHJVM01

Java Resources To Debug
Type             ==> E       (J=Java Applications, E=Enterprise Beans, C=Corba)

Class (Java Applications or Corba)
==>
==>
==>
==>



PF1=Help 3=Exit 8=Forward 12=Return
```

```
CADP      -    CICS Application Debugging Profile Manager   -     IYK2Z2T1

View Java Debugging Profile EJB      for $EXAMPLE

Bean (Enterprise Beans only)
==> example_bean_name
==>
==>
==>

Method (Enterprise Beans or Corba)
==> example_method_name
==>
==>
==>









PF1=Help 3=Exit 7=Back 12=Return
```

Figure 11. The View Java debugging profile screen, showing example profile EJB

The information displayed is described in "The Create Java debugging profile screen" on page 49.

Related tasks

"Viewing a debugging profile with the 3270 interface" on page 42

Related reference

"The List debugging profiles screen" on page 37

Function keys for the View Java debugging profile screen

The function keys for the "View Java debugging profile" screen are:

PF1
: Display the help screen

PF3
: End the debugging profile manager

PF7
Scroll forward
PF8
Scroll backward
PF12
Return to the "List debugging profiles" screen

Setting the display device

About this task

When you have created a debugging profile for a compiled language program, but before you can start debugging the application programs defined in the profile, you must specify the display device with which you will interact with the debugger. You can use:

a 3270 terminal
a debugging tool on a workstation.

The display device becomes associated with a debugging profile when you activate the profile, and remains associated with the profile until you make the profile inactive.

You can choose when you specify the display device:

- You can specify the display device before you make a profile active. The same display device will be associated with each profile that you subsequently make active.
- You can specify the display device when you make a profile active. The same display device will be associated with the profile that is made active. If you make more than one profile active at the same time (by selecting a number of profiles on the "List debugging profile" screen) the same display device will be associated with them all.

Related reference

"The List debugging profiles screen" on page 37

"The Set LE debugging display device screen" on page 56

Specifying the display device before you make a profile active

Starting with the "List debugging profiles" screen, follow these steps:

1. Press PF9 to display the "Set LE debugging display device" screen.
2. Complete the details of the display device that you want to associate with the profile.
3. Type "No" in the "Display this panel on each LE profile activation" field.
4. Press ENTER. CICS saves the display device settings, and displays the "List debugging profiles" screen.

The "Set LE debugging display device" screen will not be displayed when you activate profiles; the settings you have supplied will be applied to all compiled language profiles when you activate them.

Specifying the display device when you make a profile active

Starting with the "List debugging profiles" screen, follow these steps:

1. Press PF9 to display the "Set LE debugging display device" screen.
2. Complete the details of the display device that you want to associate with the profile.
3. Type "Yes" (the default value) in the "Display this panel on each LE profile activation" field.

4. Press ENTER. CICS saves the display device settings, and displays the "List debugging profiles" screen.

CICS will display the "Set LE debugging display device" screen whenever you activate profiles for compiled language programs. The screen is initialized with the last settings you supplied. To change the settings:

1. Change any details of the display device that you want to associate with the set of profiles that is being activated.
2. Press ENTER. The display device settings are saved, and associated with the profile. CICS displays the "List debugging profiles" screen.

The Set LE debugging display device screen

Use the "Set LE debugging display device" screen to specify the display device with which you will interact with the debugger.

```
CADP      -    CICS Application Debugging Profile Manager   -     IYK2Z2T1

Set LE Debugging Display Device

Debugging Display Device
Session Type             ==>  3270                        (3270,TCP)
3270 Display Terminal    ==>  TC15

TCP/IP Name Or Address
==>
==>
==>
==>
Port                     ==>  08000

Type of socket communication ==> Single                   (Single,Multiple)


Display this panel on LE profile activation ==> YES




Enter=Save and return PF1=Help 3=Exit 12=Cancel
```

Figure 12. The CADP Set LE debugging display device screen

Related tasks

"Activating debugging profiles with the 3270 interface" on page 41

"Setting the display device" on page 55

Related reference

"The List debugging profiles screen" on page 37

Fields on the Set LE debugging display device screen

The fields on the "Set LE debugging display device" screen are:

Session type
Specifies how you will interact with Debug Tool:

3270
You will interact with Debug Tool using a 3270 type terminal. Specify the terminal id in the display id field. This is the default value.

TCP
You will interact with Debug Tool using a debugging client on your

workstation. The client will communicate with Debug Tool using TCP/IP. Specify the port number at which the client listens for a connection in the port field.

3270 display terminal
When the session type is 3270, specify the terminal id of the terminal with which you will interact with Debug Tool. The default value is the id of the terminal at which you running CADP.

Important: The terminal specified here is not necessarily the terminal at which the transaction being debugged will run.

TCP/IP name or address
When the session type is TCP, specify the IP address or name of the host where the debugging client is running.

Port
When the session type is TCP, specify the port number at which the debugging client listens for a connection. Specify a value in the range 0 - 65535. The default is 8000.

Type of socket communication
When the session type is TCP, specifies whether the debugging client and debugging server will communicate using a single socket or more than one socket.

Single
Use a single socket for communication. This is the default value, and is the preferred value when you use a WebSphere Studio product as your debugging client.

Multiple
Use more than one socket for communication. You must specify this value when you use a VisualAge product as your debugging client.

Display this panel on LE profile activation
Specifies whether you want to display the "Set LE debugging display device" screen whenever you activate debugging profiles for compiled language programs:

YES Display the "Set LE debugging display device" screen whenever debugging profiles are activated. This is the default behavior.

NO Do not display the "Set LE debugging display device" screen whenever debugging profiles are activated. The display device that you specify will be associated with all the profiles that you activate.

Function keys for the Set LE debugging display device screen

The function keys for the "Set LE debugging display device" screen are:

PF1
Display the help screen

PF3
End the debugging profile manager

PF12
Return to the "List debugging profiles" screen

Chapter 3. CBAM - BTS browser

Use CBAM to browse the CICS business transaction services (BTS) objects (process-types, processes, activities, containers, events and timers) known to this region.

CBAM is described in CBAM BTS browser the *CICS Business Transaction Services* manual.

© Copyright IBM Corp. 1982, 2011

Chapter 4. CCRL - certificate revocation lists transaction

Use the CCRL transaction to create and update the certificate revocation lists (CRLs) that are stored in an LDAP server. Use CCRL if you are implementing SSL in your CICS regions and want each connection checked for a revoked certificate during the SSL handshake.

The CCRL transaction specifies the location of CRL repositories on the World Wide Web. CICS downloads the lists from the CRL repository at the specified URL and stores it in the LDAP server. You can specify more than one URL if you need to access multiple CRL repositories from different certificate authorities.

Before you run the CCRL transaction, you must have the following set up in CICS:

- An LDAP server that is set up and configured to store the certificate revocation lists. See the *CICS RACF Security Guide* for more information.
- The **CRLPROFILE** system initialization parameter defined with the profile that authorizes CICS to access the LDAP server that stores the certificate revocation lists.

You can run the CCRL transaction from a terminal or from a START command. If you want to schedule regular updates, use the START command option.

To run the transaction from a terminal, ensure that the terminal accepts mixed case so that you can enter the URLs.

1. Enter the following command: `CCRL` *`url-list`* where *url-list* is a space-delimited list of URLs that contain the certificate revocation lists that you want to download. For example, you could specify:

   ```
   CCRL  http://crl.CertificateAuthority.com/CRLList1.crl
       http://crl.CertificateAuthority.com/CRLList2.crl
   ```

2. Enter the administrator distinguished name and password for the LDAP server. This allows CICS to update the LDAP server with the CRLs that it downloads. The name and password are set in the file `slapd.conf` in the LDAP server.

To run the transaction from a START command, using the following syntax: `EXEC CICS START TRANSID(CCRL) FROM (admin://`*`adminDN:adminPW url-list`*`) LENGTH (`*`url-list-length`*`) [INTERVAL(`*`hhmmss`*`)|TIME(`*`hhmmss`*`)]`, where *url-list* is a space-delimited list of URLs that contain the certificate revocation lists that you want to download, *url-list-length* is the length of the URL list, and *hhmmss* is the interval or expiration time at which the CCRL transaction is scheduled to run. For example you could specify:

```
EXEC CICS START TRANSID(CCRL)
          FROM ('admin://cn=ldapadmin:cics31ldap
       http://crl.CertificateAuthority.com/CRLList1.crl
       http://crl.CertificateAuthority.com/CRLList2.crl')
        LENGTH(124) INTERVAL(960000)
```

This example schedules the CCRL transaction to run in 96 hours. If you enter an invalid URL, you receive an error message.

© Copyright IBM Corp. 1982, 2011

Chapter 5. CDBC - database control menu

Use CDBC to connect CICS to DBCTL, or to disconnect CICS from DBCTL. CDBC is applicable only to CICS systems using the IMS Database Control (DBCTL) interface.

For detailed guidance on using the CDBC transaction with databases controlled by DBCTL, see IMS Database Control (DBCTL) overview in the IMS Database Control Guide.

You can start CDBC from any terminal, from a card reader/line printer (CRLP) or DASD sequential terminal, or from the operating system console.

There are two ways of starting this transaction:

- Type the identifier, CDBC, on the command line of your display, followed by the required keyword, which can be either:

  ```
  CDBC CONnect [SUFfix(startup)]
  ```

  where **startup** is a two-digit startup table suffix, or:

  ```
  CDBC DISconnect [IMMediate]
  ```

- Type the identifier, CDBC, on the command line of your display, and press the ENTER key. You get the following display:

```
CDBC               CICS-DBCTL CONNECTION/DISCONNECTION                93.259
                                                                     13:33:20

          Select one of the following:

               1 Connection
               2 ORDERLY disconnection
               3 IMMEDIATE disconnection


          Option Selection     ==>
          Startup Table Suffix ==> 00
          DBCTL ID Override    ==>



  Status of the Interface: DFHDB8290I DBCTL not connected to CICS.
              CICS APPLID: IYAHZCD2
                 DBCTL ID:
     Startup Table Suffix:


PF1 = Help   2 = Refresh   3 = End
```

Figure 13. CDBC transaction: initial screen

As you can see from this display, you can select a CICS-DBCTL connection, an orderly disconnection, or an immediate disconnection.

This chapter describes:

- "Connecting CICS to DBCTL" on page 64
- "Disconnecting CICS from DBCTL" on page 65

© Copyright IBM Corp. 1982, 2011

Connecting CICS to DBCTL

About this task

You can connect CICS to DBCTL by either:

Procedure

- Typing CDBC CONnect [SUFFIX(xx)]
- Typing '1' in the Option Selection field indicated by ==>. You can also specify a startup table suffix in the Startup Table Suffix field. If you leave this field blank you get the existing suffix, if one exists.

Results

If you want to proceed with the CICS-DBCTL connection, press PF5 as requested on the display. The "status" line of the display is then updated. (You can find out the current status of the connection at any time by pressing PF2, which refreshes the display.)

You should get the following message:

```
DFHDB8207D DBCTL Connection requested. Press PF5
    to confirm.
```

This display also shows you the application identifier of the CICS system, and the identifier of the DBCTL system to which CICS is connected. The DBCTL identifier is available only when the connection has been made. You can also see the suffix of the startup table for this connection.

Example

The following display shows you the effect of asking for option 1:

```
CDBC               CICS-DBCTL CONNECTION/DISCONNECTION                93.259
                                                                     13:33:59

          Select one of the following:

               1 Connection
               2 ORDERLY disconnection
               3 IMMEDIATE disconnection


          Option Selection     ==> 1
          Startup Table Suffix ==> 00
          DBCTL ID Override    ==>

  DFHDB8207D Connection to DBCTL requested. Press PF5 to confirm.

  Status of the Interface: DFHDB8290I DBCTL not connected to CICS.
              CICS APPLID: IYAHZCD2
                 DBCTL ID:
     Startup Table Suffix:


PF1 = Help   2 = Refresh   3 = End
```

Figure 14. CDBC transaction: option 1 screen

Disconnecting CICS from DBCTL

About this task

To disconnect CICS from DBCTL, you can either:

Procedure

- Type

  ```
  CDBC DISConnect [IMMediate]
  ```

- Choose either option 2 or option 3 of the CDBC panel.

Results

Option 2 gives you an orderly disconnection. An orderly disconnection allows all existing CICS-DBCTL tasks to be completed before CICS is disconnected from DBCTL.

The following display shows you the effect of asking for option 2:

```
CDBC               CICS-DBCTL CONNECTION/DISCONNECTION                 93.259
                                                                      13:39:20

          Select one of the following:

               1 Connection
               2 ORDERLY disconnection
               3 IMMEDIATE disconnection


          Option Selection     ==> 2
          Startup Table Suffix ==> 00
          DBCTL ID Override    ==>

  DFHDB8209D DBCTL orderly disconnection requested. Press PF5 to confirm.

  Status of the Interface: DFHDB8293I DBCTL connected and ready.
              CICS APPLID: IYAHZCD2
                 DBCTL ID: SYS2
     Startup Table Suffix: 00


PF1 = Help   2 = Refresh   3 = End
```

Figure 15. CDBC transaction: option 2 screen

Option 3 gives you an immediate disconnection. This allows only current DL/I requests to DBCTL from this CICS system to be completed before CICS is disconnected from DBCTL.

Specifying a DBCTL system identifier (DBCTLID)

About this task

A field on the CDBC panel enables you to use the CDBC transaction to specify a DBCTLID to override the one in the DRA startup table. The syntax for connection to DBCTL using CDBC is:

```
CDBC CONNECT [SUFFIX(xx)] [DBCTLID(yyyy)]
```

For more details on using CDBC, see IMS Database Control (DBCTL) overview in the IMS Database Control Guide.

Chapter 6. CDBI - database control inquiry

Use CDBI to inquire about the status of the interface between CICS and DBCTL. You can start CDBI from 3270 (and 3270-like) terminals or from the operating system console.

For detailed information about the CICS-DBCTL interface and guidance on using the CDBI transaction with databases controlled by DBCTL, see IMS Database Control (DBCTL) overview in the IMS Database Control Guide.

Start the transaction by typing the identifier, CDBI, on the command line of your display, and press the ENTER key. You get the following display:
This display tells you the current status of the CICS-DBCTL connection, which can

```
CDBI                    CICS-DBCTL INTERFACE INQUIRY          91.184
                                                             11:23:50




           Status     : xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx.
           CICS APPLID: DBDCCICS
           DBCTL ID   : SYS1









PF1 = Help   2 = Refresh   3 = End
```

Figure 16. CDBI transaction: initial screen

be one of the following:

```
DFHDB8290I DBCTL not connected to CICS.
DFHDB8291I DBCTL connect phase 1 in progress.
DFHDB8292I DBCTL connect phase 2 in progress.
DFHDB8293I DBCTL connected and ready.
DFHDB8294I DBCTL orderly disconnect in progress.
DFHDB8295I DBCTL immediate disconnect in progress.
DFHDB8296I DBCTL cannot be connected to CICS.
```

The display also shows you the application identifier of the CICS system (in this case, DBDCCICS), and the identifier of the DBCTL system to which CICS is connected.

© Copyright IBM Corp. 1982, 2011

Chapter 7. CDBM - database control interface

Use the CDBM transaction to issue DBCTL operator commands from a CICS terminal. CDBM is applicable only to CICS systems using the IMS Database Control (DBCTL) interface.

CDBM also provides a means of maintaining a command file which stores commands. You can store commands for any reason, but repeated reuse is likely to be a frequent purpose. These stored commands can include more databases than the operator transaction panel has space for.

To use CDBM you must:

1. Have a DBCTL system running IMS V9.1 or later
2. DBCTL must have the **AOIS** parameter set to a value other than *N*. If values *A*, *C* or *R* are used, appropriate security controls must be used to allow the CICS-DBCTL interface to operate.
3. Generate and add to the DBCTL system, a PSB named DFHDBMP.

For detailed information about the CICS-DBCTL interface and guidance on using the CDBM transaction with databases controlled using DBCTL, see IMS Database Control (DBCTL) overview in the IMS Database Control Guide.

Note: The CDBM transaction uses basic mapping support (BMS) and therefore this function is available only through those terminal devices that are supported by BMS.

Using CDBM Syntax

About this task

The syntax of the CDBM transaction is:

```
CDBM /DBCTL operator command
```

where / is the default command recognition character (CRC) for DBCTL. In this case, the default CRC is the one that must be used.

Note:

1. IMS requires that each command be prefixed with the default CRC. The CRC is present only for syntax checking reasons; it does not determine the DBCTL to which the command is sent. You cannot use a CRC value to route a command to a particular DBCTL system through CDBM - it can only be sent to the one currently connected to CICS. This DBCTL may have its own CRC value which is different from the default one of /. This does not matter to CDBM however, because the / character is used only for syntax checking, and the command is presented to the connected DBCTL without a CRC, using the AIB interface
2. To ensure that the CDBM command succeeds, you might need to set the correct **AOIS** execution parameter. For example, AOIS=S ensures that the command authorization is bypassed; with any other value, the command could fail the command authorization check.

•

You can enter a DBCTL command, for example:

© Copyright IBM Corp. 1982, 2011

```
CDBM /DISPLAY  DB  DEPT*
```

or a group command, for example:

```
CDBM /GROUP  SAMPLE  STO
```

in this way.

- You can also start the transaction by typing the identifier, CDBM, on the command line of your display, as follows:

  ```
  CDBM
  ```

  Press the ENTER key. You get the panel shown in Figure 17.

```
CDBM                 CICS-DBCTL Operator Transaction              98.135
                                                                 13:24:20

Type IMS command.
             ____________________________________________________________
             ____________________________________________________________
             ____________________________________________________________
             ____________________________________________________________


For /DBDUMP or /DBRECOVER commands

Choose one. 1 1. Do not force end of volume
              2. Force end of volume


Press enter to display responses.


CICS APPLID DBDCCICS
DBCTL ID    SYS3


F1=Help  F2=Maintenance  F3=Exit  F5=Refresh  F12=Cancel
```

Figure 17. CICS-DBCTL operator transaction

On this panel you can enter a DBCTL command, for example:

```
/DISPLAY  DB  ALL
```

or a group command, for example:

```
/GROUP  SAMPLE  STA
```

An example of the use of a /GROUP command from the CICS-DBCTL Operator Transaction screen is shown in Figure 18 on page 71.

```
CDBM                CICS-DBCTL Operator Transaction              98.135
                                                                13:24:20

Type IMS command.
            /GROUP SAMPLE STA_____________________________________________
            _____________________________________________________________
            _____________________________________________________________
            _____________________________________________________________


For /DBDUMP or /DBRECOVER commands

Choose one. 1 1. Do not force end of volume
              2. Force end of volume


Press enter to display responses.


CICS APPLID DBDCCICS
DBCTL ID    SYS3


F1=Help  F2=Maintenance  F3=Exit  F5=Refresh  F12=Cancel
```

Figure 18. CICS-DBCTL operator transaction panel showing a GROUP command

Note: The group command *SAMPLE STA* referred to in this panel can be found in the "Using DFHDBFK - The CDBM GROUP command file" on page 72.

Responses to commands issued using CDBM are returned in a screen like the one shown in Figure 19, which shows the first of a number of screens issued in response to a /DISPLAY DB ALL command.

```
CDBM                     CICS-DBCTL IMS Responses             Screen 1
                                              Responses 1    to 18
                                                            More:   +
DATABASE  TYPE  TOTAL UNUSED  TOTAL UNUSED ACC  CONDITIONS
ACCUNTDB                                    UP   STOPPED, NOTOPEN, NOTINIT
ADMIDX1                                     UP   STOPPED, NOTOPEN, NOTINIT
ADMOBJ1                                     UP   STOPPED, NOTOPEN, NOTINIT
ADMOBJ2                                     UP   STOPPED, NOTOPEN, NOTINIT
ADMOBJ3                                     UP   STOPPED, NOTOPEN, NOTINIT
ADMSYSDF                                    UP   STOPPED, NOTOPEN, NOTINIT
BE1CHKPT  DL/I                              UP   NOTOPEN
BE1PARTA                                    UP   STOPPED, NOTOPEN, NOTINIT
BE1PARTB                                    UP   STOPPED, NOTOPEN, NOTINIT
BE1PARTC                                    UP   STOPPED, NOTOPEN, NOTINIT
BE1PARTS                                    UP   STOPPED, NOTOPEN, NOTINIT
BE2ORDER  DL/I                              UP   NOTOPEN
BE2ORDRX  DL/I                              UP   NOTOPEN
BE2PARTS  DL/I                              UP   NOTOPEN
BE2PCUST  DL/I                              UP   NOTOPEN
BE3ORDER  DL/I                              UP   NOTOPEN
BE3ORDRX  DL/I                              UP   NOTOPEN
                                                                More...

F1=Help  F3=Exit  F4=Top  F6=Bottom  F7=Bkwd  F8=Fwd  F9=Retrieve  F12=Cancel
```

Figure 19. CICS-DBCTL IMS responses

Using DFHDBFK - The CDBM GROUP command file

Before you begin

Before you can use the **GROUP** command, CDBM requires a file in which all your predefined commands can be stored. This file, DFHDBFK, is the CDBM GROUP command file. It is a VSAM KSDS. The DFHDBFK file is not required until you first attempt to use the /GROUP command.

About this task

You can use whatever means you prefer to define this file and to maintain its content.

Procedure

1. To define the file DFHDBFK and optionally to place some of your group definitions in it, use the sample JCL that is described in the *CICS System Definition Guide*. The example given there also shows you how to batch load this file if you want to do so. You can load many definitions in this way, or load none.
2. To use the /GROUP command functions of the CDBM transaction, add the data set name of the group command file to the definition of the DFHDBFK file within the DFHDBCTL group. If you require more information, see the *CICS Resource Definition Guide*.

What to do next

One method of maintaining the content once the file is defined is provided by CDBM which offers a Maintenance function for the DFHDBFK file.

Record layout in the CDBM GROUP command file

Each record in the DFHDBFK file may be up to 1428 characters long, as follows:

field	length	content	description
1	12	Group	a 12-character field containing your chosen name for this group. The acceptable characters are A-Z 0-9 $ @ and #. Leading or embedded blanks are not allowed, but trailing blanks are acceptable.
2	10	IMS Command	a 10-character field containing any of the IMS command verbs that are valid for CDBM (see IMS Database Control (DBCTL) overview in the IMS Database Control Guide for details). Leading or embedded blanks are not allowed, but trailing blanks are acceptable. **Note:** The validity of the IMS command verb is not checked by CDBM. Invalid values will be reported by IMS when the command is attempted.

field	length	content	description
3	1406	IMS Command parameters	Up to 1406 characters of parameters appropriate to the chosen IMS command verb. (This will often consist of lists of databases.) **Note:** Wildcard characters may not be used in the parameters stored in the CDBM Group command file. This is unlike the other functions of the CDBM transaction which permit the use of wildcard characters to describe multiple similarly named databases.

The MAINTENANCE panel for DFHDBFK

If you press the Maintenance key (PF2) on the main CDBM panel, you get the panel shown in Figure 20.

```
CDBM               CICS/DBCTL COMMAND GROUP MAINTENANCE
_        ACTION      A add  B browse  D delete  R read  U update
____________ GROUP          __________ IMS COMMAND
>                                                                  <
>                                                                  <
>                                                                  <
>                                                                  <
>                                                                  <
>                                                                  <
>                                                                  <
>                                                                  <
>                                                                  <
>                                                                  <
>                                                                  <
>                                                                  <
>                                                                  <
>                                                                  <
>                                                                  <
>                                                                  <
>                                                                  <
>                                                                  <
>                                                                  <

F1=Help  F3=Exit  F12=Cancel
```

Figure 20. CICS-DBCTL Group Maintenance panel

Input fields

The input fields are:

- Action
- Group
- IMS Command
- IMS Command parameters (between the > < marks).

Group, IMS Command and IMS Command parameters are described in the section "Record layout in the CDBM GROUP command file" on page 72

The Action field will accept one of the following:

A Add

Add a new record to the DFHDBFK file. If the key already exists, the Add fails.

Note: To Add a record that is very similar to an existing record, but which has a different key, you may find it helpful to Read the existing record, modify the displayed fields, and then Add this new record.

B Browse

Displays the contents of the command file, record by record. Specify any key (or none) to indicate where you want the browse to start. Each time you press ENTER, Browse moves on to the next record. At the end of the file you will be prompted to wrap around to the start of the file. You can accept this or not as you prefer. Incomplete keys, and unknown keys are also acceptable as start points. If no key is provided, the browse starts at the first record in the file.

If you have used Browse to locate a specific record for deletion or for update, remember to use Read before either Delete or Update.

D Delete

Delete a record from the DFHDBFK file. A Delete must be immediately preceded by a Read to lock the required record.

R Read

Read displays a specific record. Unlike Browse it does not operate on partial, or absent keys, and does not present the next record when you press ENTER.

Read is required before those actions (Delete and Update) which change an existing record. It locks that record against the possibility of being changed by another operator. This action also serves to help you confirm that the correct record has been selected.

A lock is released by exiting the CDBM maintenance panel, or by your next CDBM Maintenance action (whether that is the Update or Delete you had contemplated, or something different entirely).

U Update

Update a record in the DFHDBFK file. An Update must be immediately preceded by a Read to lock the required record.

You cannot update the key fields (GROUP and IMS COMMAND).

Note: Use Add to create a new key.

Note: In the descriptions above, *Key* refers to the 22 characters at the beginning of each record in the DFHDBFK file (namely the GROUP and IMS COMMAND).

Chapter 8. CEBR - temporary storage browse

Use the CEBR transaction to browse temporary storage queues and to delete them. You can also use CEBR to transfer a transient data queue to temporary storage in order to look at its contents and to re-create the transient data queue when you have finished.

The CEBR commands that perform these transfers allow you to create and delete transient data queues as well. Remember that:

- Browsing temporary storage queues retrieves the next record, following whichever record has most recently been retrieved by any active task.

  If you are working with the same temporary storage queue at the same time as another task, you might not retrieve the next record.
- Transient data queue reads are destructive. If you read a transient data queue that is being used on an active system by applications, it is likely to cause problems.

You start the CEBR transaction by entering the transaction identifier and then the name of the queue you want to browse. For example, to display the temporary storage queue named CEBRS209, you enter `CEBR CEBRS209`. If the queue name includes lowercase characters, ensure that uppercase translation is suppressed for the terminal you are using and then enter the correct combination of upper and lowercase characters.

CICS responds with a display of the queue, a sample of which is shown in Figure 21:

```
CEBR  TS QUEUE CEBRS209 SYSID C142 RECORD     1 OF     3    COL     1 OF    22
ENTER COMMAND ===>
     *************************  TOP OF QUEUE  ************************
00001 0000550012340000001S209
00002 0000560034560000002S209
00003 0001020005640000001S209
     ************************  BOTTOM OF QUEUE  ***********************









PF1 : HELP              PF2 : SWITCH HEX/CHAR     PF3 : TERMINATE BROWSE
PF4 : VIEW TOP          PF5 : VIEW BOTTOM         PF6 : REPEAT LAST FIND
PF7 : SCROLL BACK HALF  PF8 : SCROLL FORWARD HALF PF9 : UNDEFINED
PF10: SCROLL BACK FULL  PF11: SCROLL FORWARD FULL PF12: UNDEFINED
```

Figure 21. Typical CEBR screen displaying temporary storage queue contents

© Copyright IBM Corp. 1982, 2011

Alternatively, you can start the CEBR transaction from CEDF. Press PF5 from the initial CEDF screen (see Figure 27 on page 100) which takes you to the working-storage screen, and then you press PF2 from that screen to browse temporary storage (that is, you invoke CEBR). CEBR responds by displaying the temporary storage queue whose name consists of the four letters 'CEBR' followed by the four letters of your terminal identifier. CICS uses this same default queue name if you invoke CEBR directly and do not supply a queue name. The result of running CEBR without a queue name or from an EDF session at terminal S21A is shown in Figure 22.

If you enter CEBR from CEDF, you return to the CEDF panel when you press PF3 from the CEBR screen.

```
CEBR  TS QUEUE CEBRS209 SYSID C142 RECORD     1 OF     0    COL     1 OF  1 0
ENTER COMMAND ===>                                                      2
************************  TOP OF QUEUE  ******************************
************************  BOTTOM OF QUEUE  ***************************

                                                                        3

TEMPORARY STORAGE QUEUE CEBRS21A            IS EMPTY                    4
PF1 : HELP              PF2 : SWITCH HEX/CHAR     PF3 : TERMINATE BROWSE 5
PF4 : VIEW TOP          PF5 : VIEW BOTTOM         PF6 : REPEAT LAST FIND
PF7 : SCROLL BACK HALF  PF8 : SCROLL FORWARD HALF PF9 : UNDEFINED
PF10: SCROLL BACK FULL  PF11: SCROLL FORWARD FULL PF12: UNDEFINED
```

Figure 22. Typical CEBR display of default temporary storage queue

1. Header
2. Command line
3. Body
4. Message line
5. Menu of options

For information about the CEBR temporary storage browse transaction and guidance on using it, see the *CICS Application Programming Guide*.

Chapter 9. CEBT - master terminal (alternate CICS)

Use CEBT to control an alternate CICS system in an extended recovery facility (XRF) environment.

You can start this transaction only at the operating system console using the **MODIFY** command. The CEBT transaction is not subject to transaction security checking, which means that any user is authorized to use CEBT. You can use RACF to control who is allowed to use the MODIFY command. For more information, see Using an MVS system console as a CICS terminal in the RACF Security Guide.

Starting a transaction at the system console is described in "Using the system console" on page 9.

Minimum abbreviation of keywords

The CEBT command accepts as few characters of a keyword as needed to identify it uniquely in the request.

An exception to this rule is the SHUTDOWN keyword, for which the minimum abbreviation is SHUT.

In the syntax displays on your screen, the minimum permitted abbreviation is shown in uppercase characters, and the rest of the keyword is shown in lowercase characters.

Minimum abbreviations might change between CICS releases because of the introduction of new commands.

CEBT INQUIRE and SET commands

The **CEBT INQUIRE** and **CEBT SET** commands allow you to query or set:

- Auxiliary trace ("CEBT INQUIRE AUXTRACE" and "CEBT SET AUXTRACE" on page 81)
- The dump data set ("**CEBT INQUIRE DUMP**" on page 79 and "CEBT SET DUMP" on page 82)
- GTF tracing ("CEBT INQUIRE GTFTRACE" on page 79 and "CEBT SET GTFTRACE" on page 83)
- Internal tracing ("**CEBT INQUIRE INTTRACE**" on page 80 and "CEBT SET INTTRACE" on page 84)
- Surveillance or 'tracking' of the active CICS system by the alternate CICS system ("**CEBT INQUIRE SURVEILLANCE**" on page 81 and "CEBT SET SURVEILLANCE" on page 85)
- Takeover from the active CICS system by the alternate CICS system ("**CEBT INQUIRE TAKEOVER**" on page 81 and "CEBT SET TAKEOVER" on page 85).

CEBT INQUIRE AUXTRACE

Return information about the status of auxiliary trace.

© Copyright IBM Corp. 1982, 2011

Description

Auxiliary trace entries are made to a pair of CICS-controlled BSAM data sets, A and B.

CEBT INQUIRE AUXTRACE returns the following information:

- Whether the auxiliary trace data set is open or closed
- Which auxiliary trace data set is active, Cur(A) or Cur(B)
- Whether an auxiliary trace is in progress (started) or not (stopped).

For background information about auxiliary tracing, see the *CICS Problem Determination Guide*.

Input

Press the Clear key and type `CEBT INQUIRE AUXTRACE` (or suitable abbreviations for the keywords).

To change various attributes, use the **CEBT SET AUXTRACE** command.

CEBT INQUIRE AUXTRACE

Displayed fields

All
Automatic switching occurs as necessary until the end of this CICS session without the need for operator intervention.

Auxiliary
Identifies this panel as relating to auxiliary trace.

Current(A|B)
Displays a one character identifier of the current auxiliary data set, which can be A or B. This value is blank if there is no auxiliary trace data set.

Next
Automatic switching occurs when the current trace data set is full, but not subsequently.

Noswitch
Switching does not occur without operator intervention.

Pause
Auxiliary tracing has stopped, but the trace data set has been left open. A subsequent START request causes trace entries to be written immediately following those that were written before the PAUSE request.

Start
Auxiliary tracing is in progress.

Stop
Auxiliary tracing has stopped.

Switch
Displays the information that an immediate switch is being made from the current auxiliary trace data set to the alternate trace data set.

CEBT INQUIRE DUMP

Retrieve information about the CICS dump data sets.

Description

CEBT INQUIRE DUMP tells you which dump data set (A or B) is currently open.

For general information about dumps, see the *CICS Problem Determination Guide*..

Input

Press the Clear key and type CEBT INQUIRE DUMP (or suitable abbreviations for the keywords).

To change various attributes, use the **CEBT SET DUMP** command.

CEBT INQUIRE DUMP

Displayed fields

Autoswitch
The switch occurs only when the active dump data set is full, but not subsequently.

Closed
The active CICS dump data set is closed.

Currentdds(*value*)
Displays a one character indicator that shows which dump data set is active. This can be A or B.

Dump
Indicates that this panel relates to a dump data set inquiry.

Noautoswitch
When the active dump data set is full, there is not an automatic switch to the inactive dump data set.

Note: Because this is a negative attribute, the field appears blank.

Open
The active CICS dump data set is open.

Switch
The currently active dump data set is to become inactive, and the currently inactive data set is to become active.

CEBT INQUIRE GTFTRACE

Show the status of CICS tracing to the MVS Generalized Tracing Facility (GTF).

Description

CEBT INQUIRE GTFTRACE shows whether CICS tracing to GTF is started or stopped.

This command is intended for use at terminals that do not support the CETR full-screen tracing transaction.

For information about GTF tracing, see the *CICS Problem Determination Guide* and the *CICS Performance Guide*.

Input

Press the Clear key and type `CEBT INQUIRE GTFTRACE` (or suitable abbreviations for the keywords).

To change any attributes, use the **CEBT SET GTFTRACE** command.

CEBT INQUIRE GTFTRACE

Displayed fields

Gtftrace
Indicates that this panel relates to a GTFTRACE inquiry.

Start
CICS tracing to GTF is in progress.

Stopped
CICS tracing to GTF is not in progress.

CEBT INQUIRE INTTRACE

Show the status of CICS internal tracing.

Description

CEBT INQUIRE INTTRACE shows whether CICS internal tracing is in progress.

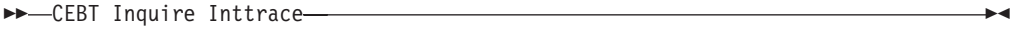
This command is intended for use at terminals that do not support the full-screen tracing transaction CETR.

Input

Press the Clear key and type `CEBT INQUIRE INTTRACE` (or suitable abbreviations for the keywords).

To change various attributes, use the **CEBT SET INTTRACE** command.

CEBT INQUIRE INTTRACE



Displayed fields

Inttrace
Indicates that this panel relates to an INTTRACE inquiry.

Start
CICS internal tracing is in progress.

Stopped
CICS internal tracing is not in progress.

CEBT INQUIRE SURVEILLANCE

Display the surveillance status of the active CICS system by the alternate CICS system.

Input

Press the Clear key and type `CEBT INQUIRE SURVEILLANCE` (or suitable abbreviations for the keywords).

To change various attributes, use the **CEBT SET SURVEILLANCE** command.

CEBT INQUIRE SURVEILLANCE

For more information about surveillance, see *CICS/ESA 3.3. XRF Guide*.

CEBT INQUIRE TAKEOVER

Display whether any takeover from the active CICS system by the alternate CICS system is to be automatic, manual, or command-driven.

Input

Press the Clear key and type `CEBT INQUIRE TAKEOVER` (or suitable abbreviations for the keywords).

To change various attributes, use the **CEBT SET TAKEOVER** command.

CEBT INQUIRE TAKEOVER

CEBT SET AUXTRACE

Function

Change auxiliary tracing options.

Description

You can start, stop, or pause tracing, and specify that automatic switching of the auxiliary data set is to occur when needed. You can use the switch option to change the data set.

Instead of using the **SET** command, you can overtype your changes on the INQUIRE screen after tabbing to the appropriate field (see "Overtyping a display" on page 117).

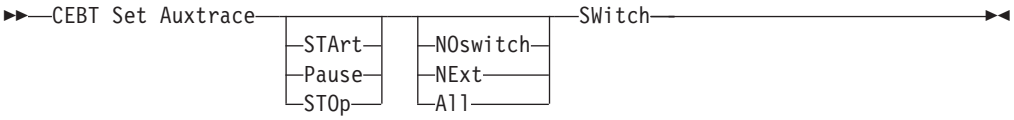
For information about using traces, see the *CICS Problem Determination Guide*.

Syntax

Press the Clear key to clear the screen. There are two ways to start this transaction:

- Type CEBT SET AUXTRACE (or suitable abbreviations for the keywords). The resulting display lists the current status, similar to that obtained by using the **CEBT INQUIRE AUXTRACE** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type CEBT SET AUXTRACE (or suitable abbreviations for the keywords), followed by one or more attribute settings that you want to change. For example, CEBT S AUX STO stops auxiliary tracing if it is already running.

CEBT SET AUXTRACE



Options

All
Specifies that automatic switching between the two auxiliary trace data sets is to occur as necessary until the end of this CICS session, without the need for operator intervention.

NExt
Automatic switching of the auxiliary trace data set is to occur when the current trace data set is full, but not subsequently.

NOswitch
Switching between the auxiliary trace data sets is not to occur without operator intervention.

Pause
Auxiliary tracing is to stop, but the data set is to remain open. A subsequent START request causes trace entries to be written immediately following those that were written before the PAUSE request.

STArt
CICS is to start auxiliary tracing and open the auxiliary trace data set if it is currently closed.

STOp
CICS is to stop auxiliary tracing and close the auxiliary trace data set. A subsequent START request causes new trace entries to be written at the start of the data set, thereby overwriting the trace entries that were written before the STOP request.

SWitch
Specifies that an immediate switch from the current auxiliary trace data set is to be made to the alternate trace data set.

CEBT SET DUMP

Change the attributes of a CICS dump data set.

Description

You can use the **CEBT SET DUMP** command to open or close the active CICS dump data set, and specify whether or not the automatic switch to the inactive dump data set is to occur when the active dump data set is full. Instead of using the SET

command, you can overtype your changes on the INQUIRE screen after tabbing to the appropriate field (see “Overtyping a display” on page 117).

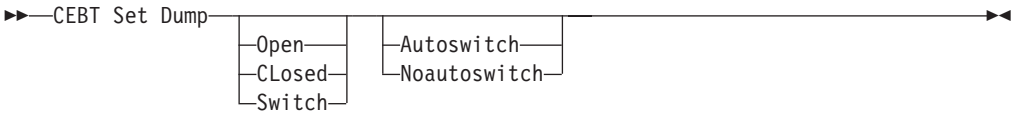
For information about using dumps, see the *CICS Problem Determination Guide*.

Syntax

Press the Clear key to clear the screen. There are two ways to start this transaction:

- Type `CEBT SET DUMP` (or suitable abbreviations for the keywords). The resulting display lists the current status, similar to that obtained by using the **CEBT INQUIRE DUMP** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEBT SET DUMP` (or suitable abbreviations for the keywords), followed by one or more attribute settings that you want to change. For example, `CEBT S DU A` causes an automatic switch to the inactive dump data set if no automatic switch was previously specified.

CEBT SET DUMP



Options

Autoswitch
The switch occurs **only** when the active dump data set is full, but not subsequently.

CLosed
The active CICS dump data set is closed.

Noautoswitch
When the active dump data set is full, there is not to be an automatic switch to the inactive dump data set.

Open
The active CICS dump data set is opened.

Switch
The currently active dump data set is to become inactive, and the currently inactive data set is to become active.

CEBT SET GTFTRACE

Change the status of MVS Generalized Tracing Facility (GTF) tracing.

Description

CEBT SET GTFTRACE allows you to switch CICS tracing to GTF on and off. This command is intended for use at terminals that do not support the CETR full-screen tracing transaction.

For information about using GTF tracing, see the *CICS Problem Determination Guide*.

Syntax

Press the Clear key to clear the screen. There are two ways to start this transaction:

- Type `CEBT SET GTFTRACE` (or suitable abbreviations for the keywords). The resulting display lists the current status, similar to that obtained by using the **`CEBT INQUIRE GTFTRACE`** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEBT SET GTFTRACE` (or suitable abbreviations for the keywords), followed by one or more attribute settings that you want to change. For example, `CEBT S G STO` stops GTF tracing if it is already running.

CEBT SET GTFTRACE

Options

`STArt`
CICS tracing to the MVS Generalized Tracing Facility (GTF) is started.

`STOp`
CICS tracing to the MVS Generalized Tracing Facility (GTF) is stopped.

CEBT SET INTTRACE

Switch CICS internal tracing on or off.

Description

This command is intended for use at terminals that do not support the full-screen tracing transaction CETR.

Syntax

Press the Clear key to clear the screen. There are two ways to start this transaction:

- Type `CEBT SET INTTRACE` (or suitable abbreviations for the keywords). The resulting display lists the current status, similar to that obtained by using the **`CEBT INQUIRE INTTRACE`** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEBT SET INTTRACE` (or suitable abbreviations for the keywords), followed by one or more attribute settings that you want to change. For example, `CEBT S IN STO` stops internal tracing if it is already running.

CEBT SET INTTRACE

Options

`STArt`
CICS internal tracing is started.

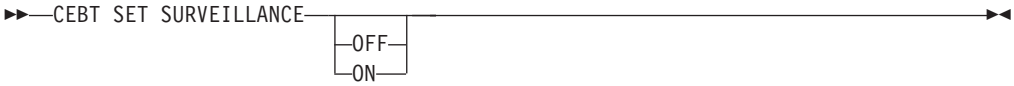
STOp
CICS internal tracing is stopped.

CEBT SET SURVEILLANCE

Set the surveillance status of the active CICS region by the alternate CICS region.

Syntax

CEBT SET SURVEILLANCE



Options

OFF
Stop surveillance of the active CICS region by the alternate CICS region.

ON
Start surveillance of the active CICS region by the alternate CICS region.

For more information about surveillance, see *CICS/ESA 3.3. XRF Guide*.

CEBT SET TAKEOVER

Set takeover from the active CICS system by the alternate CICS system to be automatic, manual, or command-driven.

Syntax

Press the Clear key to clear the screen. There are two ways to start this transaction:

- Type `CEBT SET TAKEOVER` (or suitable abbreviations for the keywords). The resulting display lists the current status, similar to that obtained by using the **CEBT INQUIRE TAKEOVER** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEBT SET TAKEOVER` (or suitable abbreviations for the keywords), followed by one or more attribute settings that you want to change. For example, `CEBT S T M` changes takeover to a manual operation if it was previously specified to be automatic.

CEBT SET TAKEOVER

Options

Automatic
Takeover is to occur automatically.

Command
Takeover is to be by command.

Manual
Takeover is to be manual.

The *CICS/ESA 3.3. XRF Guide* tells you more about these command options.

CEBT PERFORM commands

You can use the **CEBT PERFORM** commands for the following tasks:

- Shut down the alternate CICS system ("**CEBT PERFORM SHUTDOWN**")
- Request a dump of the alternate CICS and continue processing ("**CEBT PERFORM SNAP**")
- Cause the alternate CICS to take over from the active CICS ("CEBT PERFORM TAKEOVER" on page 87).

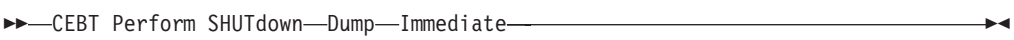
CEBT PERFORM SHUTDOWN

Shut down the alternate CICS system in an XRF complex.

Description

If you enter **CEBT PERFORM SHUTDOWN**, all active system tasks are allowed to finish, and the alternate CICS system signs off normally from the CICS availability manager (CAVM) data set.

CEBT PERFORM SHUTDOWN



Options

Dump
Produces a dynamic storage dump when the shutdown is complete.

Immediate
Shuts down the alternate CICS system immediately, terminating all active tasks. The alternate CICS signs off abnormally from the CAVM data set. This action can cause the primary (active) CICS system to abend if tasks are still running.

CEBT PERFORM SNAP

Produce a snapshot dump of the alternate CICS system while CICS continues to run.

Description

CEBT PERFORM SNAP causes a snapshot dump of the alternate CICS system to be taken while CICS continues to run. All **CEBT PERFORM SNAP** requests result in an SDUMP on an MVS SYS1.DUMP*nn* data set. The use of **CEBT PERFORM SNAP** prevents all other tasks from running on the alternate system, and could delay takeover, or recovery after takeover.

Syntax

CEBT PERFORM SNAP

CEBT PERFORM TAKEOVER

The **CEBT PERFORM TAKEOVER** command initiates a takeover by the alternate CICS system, causing the active CICS system to sign off abnormally.

Syntax

CEBT PERFORM TAKEOVER

Options

Dump
Sends a message to the active CICS system requesting it to take a system dump from the CAVM data set. This request is not honored if the active CICS fails before the dump can be taken.

Chapter 10. CECI - command-level interpreter

Use CECI, which invokes the CICS command-level interpreter, to enter an **EXEC CICS** command, check its syntax, and modify it if required. In most cases, you can also process the command.

For programming information on each **EXEC CICS** command, see the *CICS Application Programming Reference* or the *CICS System Programming Reference*.

For details of how to use the CECI transaction, see the *CICS Application Programming Guide*, which includes an example of a complete session with CECI.

Start the transaction by entering CECI on the command line of the screen, as follows:

```
CECI
```

Press ENTER to get a list of all **EXEC CICS** commands. If you then enter the command or its abbreviation on the top line and press ENTER, you get a complete display of the syntax for that command.

Alternatively, you can enter CECI together with the name of the command in which you are interested and bypass the full list of commands; for example:

```
CECI asktime
```

In both cases, you can usually go on to check the syntax of the command, change it if required, and then process the command.

Many CICS system programming commands use CVDA values and CECI shows numeric equivalents for these. See the *CICS System Programming Reference* for programming information, including a translation of these numeric values to their alphabetic values.

Command-level syntax-checking transaction (CECS)

Use the CECS transaction, which also invokes the command-level interpreter, to check the syntax of an **EXEC CICS** command, but not to process it.

Start the transaction by entering CECS on the command line of the screen, as follows:

```
CECS
```

Press ENTER. You get a list of all the **EXEC CICS** commands. Enter the command or its abbreviation at the top of the screen (next to the ?) and press ENTER and you get a complete display of the syntax for that command.

Alternatively, you can enter CECS together with the name of the command in which you are interested, and bypass the full list of commands:

```
CECS command
```

For information how to use the CECI and CECS transactions, see the *CICS Application Programming Guide*.

© Copyright IBM Corp. 1982, 2011

Chapter 11. CEDA - resource definition online

Use CEDA to define resources to your CICS system while the system is running. With CEDA, you can update both the CICS system definition (CSD) file and the running CICS system.

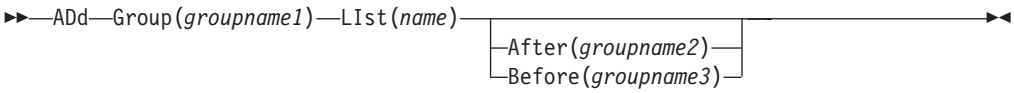
This section briefly describes CEDA and it's syntax. CEDA is described in detail in the *CICS Resource Definition Guide*.

Command syntax

Consult the *CICS Resource Definition Guide* for information about the attributes applicable to each resource type.

CEDA

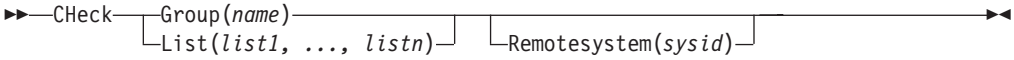
ADD



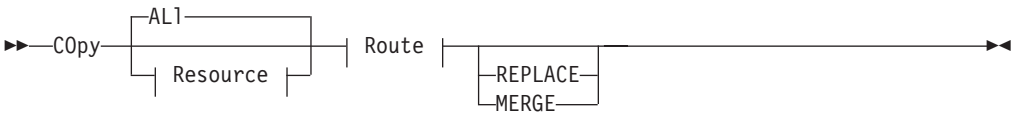
ALTER

APPEND

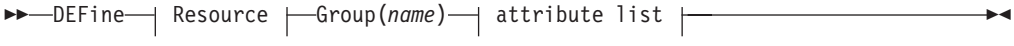
CHECK



COPY

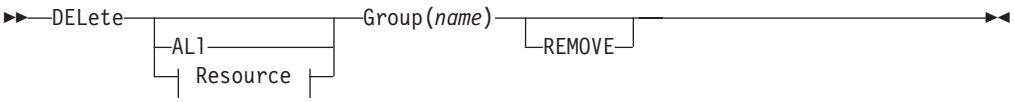


DEFINE

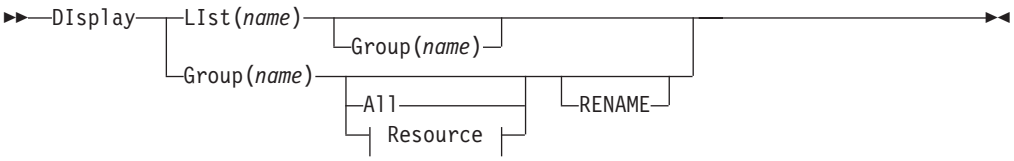


© Copyright IBM Corp. 1982, 2011

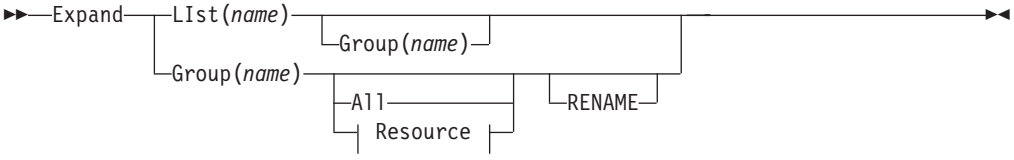
DELETE



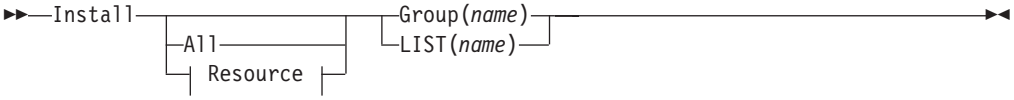
DISPLAY



EXPAND

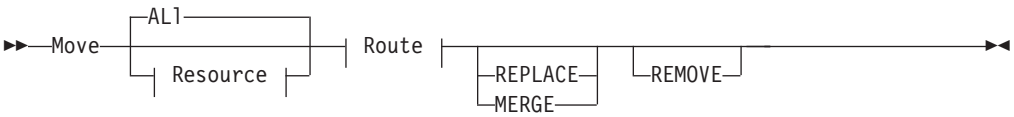


INSTALL

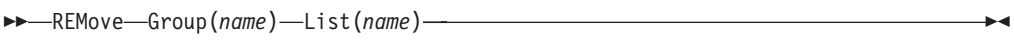


LOCK

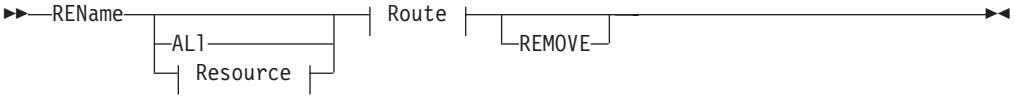
MOVE



REMOVE

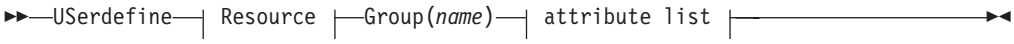


RENAME

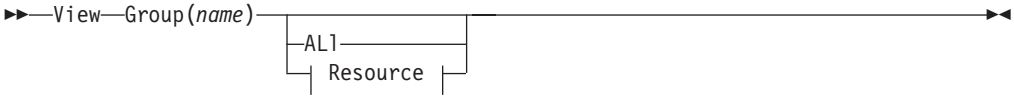


UNLOCK

USERDEFINE

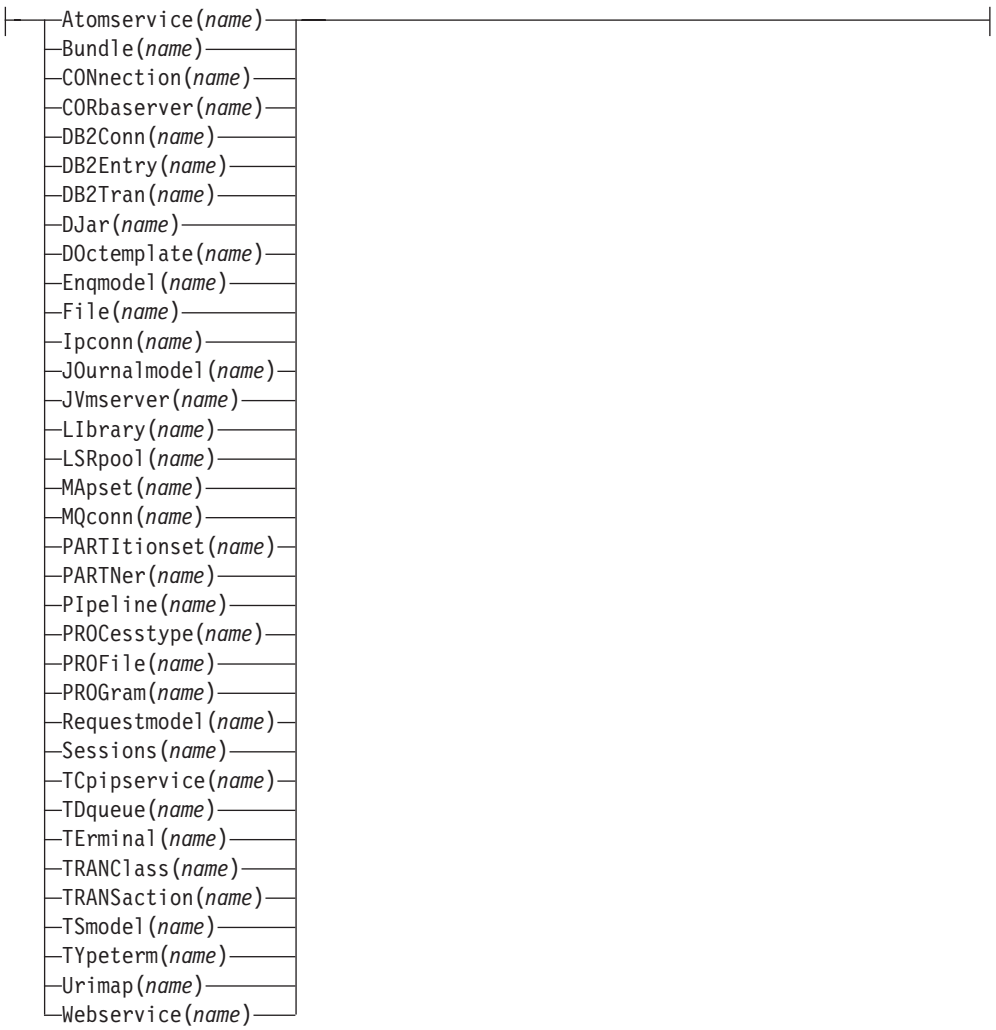


VIEW

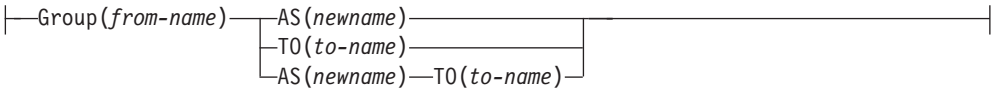


CEDA continued

Resource:



Route (Copy and Move):



Route (Rename):

Attribute list:

You can go directly to the working screen to make your changes by typing:
`CEDA function resource_type`

For example:

```
CEDA DEF CONN
```

CSD compatibility panel

Figure 23 shows a normal ALTER panel for the bottom part of the REQUESTMODEL definition, with the OMGINTERFACE, OMGMODULE, and OMGOPERATION attributes, that are now obsolete, protected as indicated by the colon symbol (:).

```
  ALTER                                                        RELEASE 0670
  OVERTYPE TO MODIFY
.
.
.
  TRANSACTION ATTRIBUTES
   TRansid      ==> CIRP

  PREVIOUS ATTRIBUTES
   OMGModule    :
   OMGInterface :
   OMGOperation :
```

Figure 23. The ALTER panel for REQUESTMODEL: normal mode

You should note that a large field, such as Module, when it is not fully displayed on the current screen, is also protected:

```
   Requestmodel   : ........
   Group          : ........
   Description  ==> ....................................
   Corbaserver  ==> *
   TYpe         ==> Generic            Corba | Ejb | Generic
  EJB PARAMETERS
   Beanname     ==> *
                ==>
                ==>
                ==>
   INTFacetype  ==> Both               Both | Home | Remote
  CORBA PARAMETERS
   Module       :
                :
```

Figure 24. The DEFINE panel for REQUESTMODEL

However, when you scroll forward, the whole field is displayed and becomes enabled:

```
CORBA PARAMETERS
 Module       ==> *
              ==>
              ==>
              ==>
 INTErface    ==> *
              ==>
              ==>
              ==>
COMMON PARAMETERS
 OPeration    ==> *
              ==>
              ==>
              ==>
TRANSACTION ATTRIBUTES
 TRansid      ==> CIRP

PREVIOUS ATTRIBUTES
 OMGModule    :
 OMGInterface :
 OMGOperation :
```

Figure 25. The DEFINE panel for REQUESTMODEL

See the *CICS Resource Definition Guide* for details of using CEDA. See the *CICS Resource Definition Guide* for background information about the CSD compatibility panel.

Use the PF2 key to change to the compatibility mode panel shown in Figure 26. The OMGINTERFACE, OMGMODULE, and OMGOPERATION attributes are now shown with the ==> symbols, indicating that they can be modified. Use the PF2 key again to return to the normal panel.

```
ALTER                                    COMPATIBILITY MODE
.
.
.
TRANSACTION ATTRIBUTES
 TRansid      ==> CIRP

PREVIOUS ATTRIBUTES
 OMGModule    ==>
 OMGInterface ==>
 OMGOperation ==>
```

Figure 26. The ALTER panel for REQUESTMODEL: compatibiity mode

Note that it is not possible to have both current and obsolete attributes in the same request model definition. If you enter OMG attribute values, you must remove all attribute values from previous versions. For more information, see the *CICS Resource Definition Guide*.

Security checking

The scope of a particular CEDA command can be limited by a combination of several levels of security checking.

The normal transaction-attach security is described in the *CICS RACF Security Guide*. CEDA calls the exec interface programs to process the user's commands, in much the same way as CECI or a user's command-level program would. The resource security and the command security used by the command-level programming interface are thus applicable to the master terminal user.

For programming information about command security checking, see the *CICS System Programming Reference*.

CEDB and CEDC

Two further resource definition transactions, CEDB and CEDC, allow you to use some, but not all, of the functions of CEDA. These transactions are described in the CEDA transaction tutorial in the *Resource Definition Guide*.

CEDB

When you use the CEDB transaction, the INSTALL command is not available to you. This means that you can update the CSD, but not the running CICS system.

CEDC

The CEDC transaction allows you only to look at data on the CICS system definition (CSD) file. You cannot update either the CSD file or the running CICS system. The only options that are available are DISPLAY, EXPAND, and VIEW.

Chapter 12. CEDF and CEDX - the execution diagnostic facility

The CICS execution diagnostic facility (EDF) provides two transactions that you can use for testing application programs.

These transactions - CEDF and CEDX - enable you to test application programs interactively without having to supply special program testing procedures.

CEDF
Use CEDF to invoke EDF for testing application programs that are associated with user transactions initiated from a terminal.

CEDX
Use CEDX to invoke EDF for testing application programs that are associated with non-terminal transactions.

Using the CEDF transaction

CEDF allows you to intercept your application programs at the program's initiation, at each CICS command, and at the program termination. CEDF helps you to isolate and focus on problems in your application programs.

About this task

For information on how to use the CEDF transaction, see the *CICS Application Programming Guide*.

You must ensure that the EDF resource definitions are installed. These resource definitions are provided in the IBM-supplied group, DFHEDF. For information about installing the resource definitions, see the *CICS Resource Definition Guide*.

You must also ensure that the terminal from which you run EDF is in transceive (ATI/TTI) status.

Command syntax

CEDF

Command options

OFF
specifies whether the EDF screen is to be switched OFF. If you specify OFF you **must** enter the preceding comma, as shown in the following examples:

```
CEDF T33A
CEDF T33A ,OFF
CEDF ,OFF
```

© Copyright IBM Corp. 1982, 2011

ON
specifies whether the EDF screen is to be switched ON. The default is ON. If you specify ON you **must** enter the preceding comma, as shown in the following examples:

```
CEDF T33A
CEDF ,ON
```

sessionid
specifies the session identifier used to test a transaction attached across an MRO, APPC, or LU6.1 session. All CICS commands processed by the attached transaction are monitored.

sysid
specifies the system identifier used to test a transaction across an MRO or APPC session.

termid
specifies the identifier of the terminal on which the transaction to be tested is being run. If you do not provide an identifier, the terminal at which you enter the CEDF transaction is assumed.

Beginning an EDF session (single-screen mode)

About this task

To start an EDF session with a single terminal, clear the screen and enter the transaction identifier, CEDF. The EDF inputs and outputs are interleaved with those of the transaction. This is known as *single-screen mode*.

```
CEDF
```

When you press ENTER, the response is:

```
THIS TERMINAL: EDF MODE ON
```

Now clear the screen and enter the name of your transaction. Figure 27 shows the initial screen for a typical transaction using EDF.

```
 TRANSACTION: ACCT  PROGRAM: ACCT00  TASK: 0000054  APPLID: 00001349  DISPLAY: 00
 STATUS:  PROGRAM INITIATION

    EIBTIME      = 110343
    EIBDATE      = 89174
    EIBTRNID     = 'ACCT'
    EIBTASKN     = 54
    EIBTRMID     = '025L'

    EIBCPOSN     = 4
    EIBCALEN     = 0
    EIBAID       = X'7D'                                    AT X'00543F1E'
    EIBFN        = X'0000'                                  AT X'00543F1F'
    EIBRCODE     = X'000000000000'                          AT X'00543F21'
    EIBDS        = '........'
+   EIBREQID     = '........'

 ENTER:  CONTINUE
 PF1 : UNDEFINED           PF2 : SWITCH HEX/CHAR     PF3 : END EDF SESSION
 PF4 : SUPPRESS DISPLAYS   PF5 : WORKING STORAGE     PF6 : USER DISPLAY
 PF7 : SCROLL BACK         PF8 : SCROLL FORWARD      PF9 : STOP CONDITIONS
 PF10: PREVIOUS DISPLAY    PF11: UNDEFINED           PF12: UNDEFINED
```

Figure 27. CEDF transaction: initial screen

Beginning an EDF session (dual-screen mode)

About this task

You can also use EDF in what is known as *dual-screen mode*; that is, using one terminal to monitor a transaction running at a second terminal.

This mode is started by adding the terminal identifier (termid) of the second terminal to the CEDF command at the first terminal:

```
CEDF tttt
```

The message that CEDF gives in response to this depends on whether there is already a transaction running on the second terminal. If the second terminal is not busy, the message displayed at the first terminal is:

```
TERMINAL tttt: EDF MODE ON
```

and nothing further will happen until a transaction is started on the second terminal, when the PROGRAM INITIATION display will appear.

If a transaction is already running on the second terminal, the message displayed at the first terminal is:

```
TERMINAL tttt: TRANSACTION RUNNING: EDF MODE ON
```

CEDF then picks up control at the next command processed and presents the appropriate display. At the first terminal, you can then follow the sequence of commands in the transaction running at the second terminal.

Monitoring remote transactions

In a multiregion operation (MRO) or an intersystem communication (ISC) environment (APPC only), you can use CEDF (in single screen mode only) for transactions that are defined in the terminal owning region (TOR) as remote. CICS automatically notifies the application owning region (AOR) that the transaction is to be run in EDF mode. When using an IPIC connection, use CEDX for transactions that are defined in the terminal owning region (TOR) as remote. IPIC does not support sending EDF information.

About this task

When the remote application ends, if your reply is YES, the terminal remains in EDF mode as usual. However, CICS deletes all the associated temporary storage queues, and none of the previous EDF screens or options is saved: you must type these again for the next transaction. Responding NO at the termination screen ends the EDF session in all participating regions.

Invoking CEBR and CECI from CEDF

About this task

Procedure

- To invoke the CEBR transaction from the CEDF screen, press PF2. You can then use CEBR commands to browse temporary storage queues and to delete them. CEBR commands are described briefly in Chapter 8, “CEBR - temporary storage browse,” on page 75

- To invoke the CECI transaction from the CEDF screen, press PF5. You can then use CECI commands to display or alter resource status. CECI commands are described briefly in Chapter 10, “CECI - command-level interpreter,” on page 89.

Results

If you are not authorized to use the CEBR or CECI transaction by the external security manager, a warning message will appear on the EDF panel. See the *CICS RACF Security Guide* for information about security checking of transactions running under CEDF.

Example

See Figure 28 for an example of the screen from which CEBR and CECI can be invoked.

```
TRANSACTION: DLID PROGRAM: DLID     TASK: 0000049 APPLID: IYAHZCIB
 ADDRESS: 00000000









WORKING STORAGE IS NOT AVAILABLE
ENTER:  CURRENT DISPLAY
PF1 : UNDEFINED           PF2 : BROWSE TEMP STORAGE PF3 : UNDEFINED
PF4 : EIB DISPLAY         PF5 : INVOKE CECI         PF6 : USER DISP
PF7 : SCROLL BACK         PF8 : SCROLL FORWARD      PF9 : STOP CONDITIONS
PF10: PREVIOUS DISPLAY    PF11: UNDEFINED           PF12: UNDEFINED
```

Figure 28. Typical EDF display from which CEBR or CECI can be invoked

What to do next

For detailed guidance on using CEBR, see in the *CICS Application Programming Guide*. For detailed guidance on using CECI, see the *CICS Application Programming Guide*.

Using the CEDX transaction

About this task

Use CEDX to monitor and debug non-terminal transactions. The transaction you specify for debugging can be:

- Initiated without an associated terminal; for example, by an EXEC CICS START command, or by a transient data queue trigger-level.
- Initiated at a terminal, which can be either the EDF terminal or a different terminal.

CICS intercepts the transaction specified on the CEDX *tranid* command, and displays the EDF diagnostic panels at the terminal at which the EDF command is issued.

CEDX provides the same function and diagnostic display panels as CEDF, and the same basic rules for CEDF also apply to CEDX. However, CICS uses the RESSEC and CMDSEC values of the CEDF transaction to authorize the use of the CEDX transaction.

Command syntax

CEDX

Command options

OFF
specifies that the EDF screen is to be switched OFF. If you specify OFF you must enter the preceding comma, as shown in the following example:

```
CEDX TRNA,OFF
```

ON
specifies that the EDF screen is to be switched ON. The default is ON. If you specify ON you must enter the preceding comma, as shown in the following example:

```
CEDX TRNB,ON
```

tranid
specifies the identifier of a transaction that you want to test using the execution diagnostic facility. The diagnostic panels are displayed on the terminal at which you issue the CEDX *tranid* command.

The transaction you specify on the CEDX command must run in the local CICS region (CRTE followed by CEDX is not allowed).

CEDX cannot interrupt transactions that are already running when you issue the CEDX command. It affects only a transaction that starts *after* you issue the command.

The following diagram illustrates the use of CEDX to invoke EDF for a transaction initiated by an EXEC CICS START command. It shows two terminals. The CICS terminal used for EDF issues a CEDX command before BBBB is started, to invoke EDF services to intercept and debug the specified transaction. The CICS user terminal issues an EXEC CICS START command to start BBBB, without a terminal.

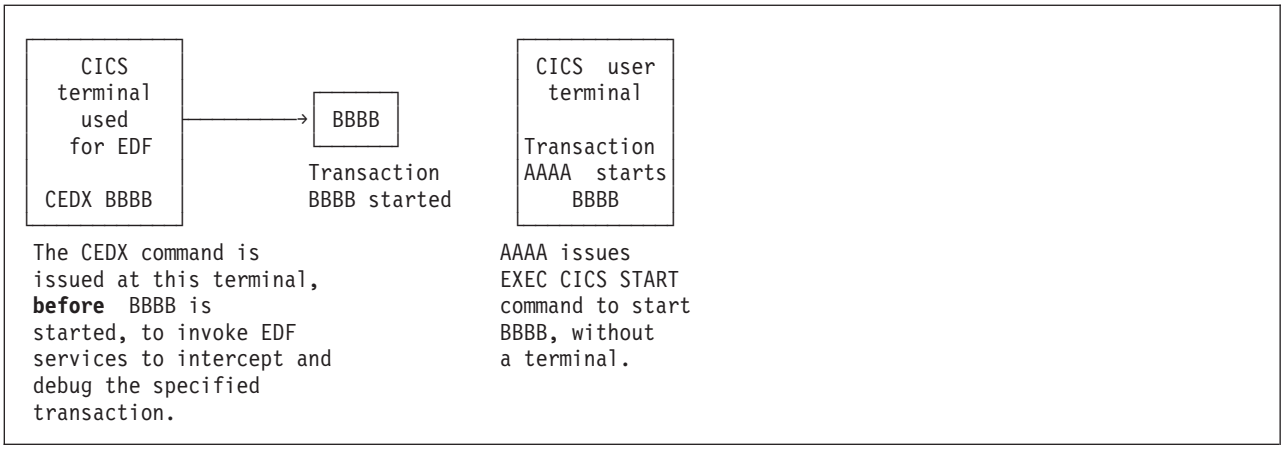


Figure 29. Invoking EDF by CEDX

Chapter 13. CEKL - master terminal (emergency use)

Use CEKL to remove tasks from a system in a situation where you cannot issue the CEMT transaction.

There are two commands, **CEKL INQUIRE TASK** and **CEKL SET TASK**. The INQUIRE command displays information about selected tasks. The SET command allows you to remove an individual task using the PURGE, FORCEPURGE, or KILL option.

Note:

1. The CEKL transaction differs from other CICS supplied transactions in that you do not have to install definitions for it.
2. You can start this command at any operating system console that has the authority to issue MODIFY commands for the relevant CICS region. This does not require a CICS terminal definition for the relevant console. You would normally issue the command from an SDSF session (using the ULOG facility to view the extended console output).

The CEKL transaction forms part of the CICS console interface and runs on a dedicated CICS internal TCB that enables commands to be processed even when the rest of CICS is unresponsive.

Minimum abbreviation of keywords

The CEKL command accepts as few characters of a keyword as are needed to identify it uniquely in the request.

In the syntax shown, the minimum permitted abbreviation is shown in uppercase characters, and the rest of the keyword is shown in lowercase characters.

CEKL INQUIRE TASK

INQUIRE TASK returns information about user tasks, and displays the list on the console from which the command was issued and the job log for the CICS region.

Description

You can enter this command in the following ways from the system console:

- `/F cicsregionname,CEKL INQUIRE TASK`

  This command displays the current status of the user tasks in your system. The minimum abbreviation is `CEKL I TA`.
- `CEKL INQUIRE TASK`, followed by one of the options to limit the range of information that you require.

  You can restrict the list of tasks by using `CEKL I TA D`. The resulting display shows the details of only those tasks that are marked for dispatch in your system.

  You can further restrict your selection by using the transaction class or transaction id option, for example, `CEKL I TA TRANClass(value)`. This command returns a list of the tasks for the specified transaction class.
- `/F cicsregionname,CEKL INQUIRE TASK(tasknum)`

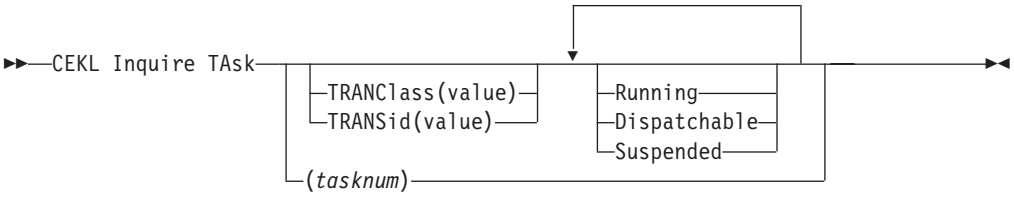
© Copyright IBM Corp. 1982, 2011

This command displays the current status of the task in your system. The minimum abbreviation is `CEKL I TA(tasknum)`.

Syntax

The syntax of the CEKL INQUIRE TASK command is as follows:

CEKL INQUIRE TASK



Options

Task(*tasknum*)
Selects the specified task. *tasknum* is a CICS-generated task number, in the range 1-99999. Task entered without a *tasknum* lists all the tasks on your system. A maximum of 1000 tasks are returned.

TRANClass (*value*)
Selects only user tasks from the specified transaction class. TRANClass is a unique transaction class name (up to 8 characters) to which the transaction belongs. You cannot use the symbols * and + to specify transaction class.

TRANSid (*value*)
Selects only user tasks with the specified transaction name (up to 4 characters) associated with the transaction. You cannot use the symbols * and + to specify transaction class.

Running
Selects tasks that are executing. These tasks are also included if you specify none of the category options (DISPATCHABLE, RUNNING, and SUSPENDED).

Dispatchable
Selects tasks that are ready to run (dispatchable). These tasks are also included if you specify none of the category options (DISPATCHABLE, RUNNING, and SUSPENDED).

Suspended
Selects tasks that are suspended (tasks waiting for some event or condition). Tasks that have not reached the point of initial dispatch either because the transaction class limit has been reached or because the maxtasks limit has been reached. These tasks are also included if you specify none of the category options (DISPATCHABLE, RUNNING, and SUSPENDED).

Results

The following example output is typical output from the inquire command.

```
TAS(nnnnn) ppp ddd cc tttt TCL(tttttttt) USE(aaaaaaaa) ATT(sssss) CPU(sssss)
RUA(sssss) HTI(sssss) HTY(rrrrrrrr) HVA(rrrrrrrr)
```

TAS(*nnnnn*)
nnnnn Displays the CICS-generated task number in the range 1–99999 for this task.

ppp *ppp* Shows whether the task is being purged, forcepurged or killed. It displays one of the following: `PUR`, `FOR`, or `KIL`. A blank implies that none of the actions have been tried.

ddd *ddd* Shows whether the task is in a dispatchable, running or suspended state. It displays one of the following: `DIS`, `RUN`, or `SUS`.

cc *cc* Specifies the TCB that is associated with this task. The values for *cc* are:

CK Indicates that the task is associated with a CICS key open TCB, for example, a JVM (mode J8) TCB.

IN Indicates that the task is associated with one of the CICS internal TCBs.

QR Indicates that the task is associated with the CICS QR TCB.

UK Indicates that the task is associated with a user key open TCB.

If the task has not reached the point of initial dispatch, the value is blank.

tttt Displays the name of the transaction (up to 4 characters) associated with the task.

TCL(*tttttttt*)
Displays the name of the transaction class (up to 8 characters) associated with the task. It the task is not defined in a class, the value is blank.

USE(*aaaaaaaa*)
Displays the userid for the task.

ATT(*sssss*)
Displays the length of time (in seconds) since the task was attached. A value of 99999 indicates that the time is greater than 24 hours.

CPUtime(*sssss*)
Displays the CPU time, in seconds, used by this task. If monitoring is inactive, the value (*****) is returned. If there is no value returne,d the task has reached either a transaction class limit or its maxtasks limit. If the task has not reached the point of initial dispatch, the value is blank. A value of 99999 indicates that the time is greater than 24 hours.

RUnAway(*sssss*)
Displays the current limit, in seconds, used by CICS to monitor the task for a runaway condition. If the value is zero, CICS is not monitoring the tasks for a runaway condition.

For suspended tasks only, the output includes HTIme(*sssss*), HTYpe(*reason*), and HVAlue(*resourcename*). The values for these three attributes are:

HTIme(*sssss*)
Displays the time (in seconds) that the task has been in the current suspended state. A value of 99999 indicates that the time is greater than 24 hours.

HTYpe(*rrrrrrrr*)
Displays the reason why the task is suspended.

HVAlue(*rrrrrrrr***)**
Displays a resource name (up to 16 characters), for example, a file name or a TS queue name.

Note: Blanks can be returned for HTYpe and HVAlue. For example, this occurs if the NAME option is not specified on the WAIT EXTERNAL command.

Sample screen

The sample output shows the output that is generated and returned to the TSO User log. In this example, the CEKL command has been used to obtain a list of all the tasks in the system MEACO22X and has returned 14 user tasks. Only the first three tasks are shown in this sample. The first two tasks are suspended and the third one is running.

The output for task 5 indicates that the task is suspended, the TCB is the QR TCB, and the transaction id is XSSY. The userid for the transaction is CICSUSER, the attach time is 6731 seconds, the amount of CPU time used is 39 seconds, and the runaway time limit is 5 seconds. The second line returns the values for the time that the task has been suspended (HTime), the reason why the task is suspended (Htype), and the associated resource name (Hvalue). The second task is similar to the first.

The third task, with the transaction id of L001, is still running in the QR TCB.

```
F MEACO22X,CEKL I TA
+IYK2ZFX3 DFHCQ0243I  CEKL INQUIRE: 00014 task(s) matched selection criteria.
+IYK2ZFX3 TAS(00005)     SUS QR XSSY              USE(CICSUSER) ATT(06731) CPU(00039) RUA(00005)
+IYK2ZFX3 HTI(06732) HTY(ICMIDNTE) HVA(DFHAPTIM)
+IYK2ZFX3 TAS(00006)     SUS QR XSSY              USE(CICSUSER) ATT(06731) CPU(00093) RUA(00005)
+IYK2ZFX3 HTI(03293) HTY(ICEXPIRY) HVA(DFHAPTIX)
+IYK2ZFX3 TAS(00033)     RUN QR LOO1              USE(CICSUSER) ATT(00033) CPU(00023) RUA(00010)
```

Figure 30. CEKL INQUIRE TASK output

The second example shows where a task number has been selected and one record is returned.

```
F MEACO22X,CEKL I TAS(62)
+IYK2ZFX3 DFHCQ0241I  CEKL INQUIRE: task number 00062 found.
+IYK2ZFX3 TAS(00062)     SUS QR XEMT              USE(CICSUSER) ATT(00176) CPU(00069) RUA(00005)
+IYK2ZFX3 HTI(00176) HTY(ZCIOWAIT) HVA(DFHZARQ1)
```

Figure 31. CEKL INQUIRE TASK output

CEKL SET TASK

Requests the cancellation of a user task.

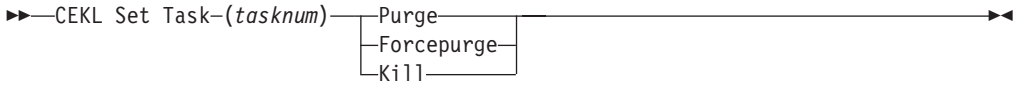
Description

CEKL SET TASK initiates a PURGE, FORCEPURGE, or KILL request.

Syntax

The syntax of the CEKL SET TASK command is as follows:

CEKL SET TASK



Options

Task(*tasknum*)
The CICS-generated task number. This is in the range 1–99999.

Forcepurge
The task is to be terminated. Data integrity is not guaranteed. Before using forcepurge, you should use purge. In some cases, for example if a task is forcepurged during backout processing, CICS terminates abnormally. If you want to terminate a task but do not want to terminate CICS, use PURGE instead of FORCEPURGE.

Kill
The task is to be terminated. System and data integrity is not guaranteed. The KILL option extends the PURGE and FORCEPURGE options. It should be used only after an attempt has been made to PURGE or FORCEPURGE a task. The KILL option does not guarantee integrity of any kind but in some situations it allows the user to free up a stalled region enabling the region to continue processing. In some cases, for example, if a task is killed during backout processing, CICS terminates abnormally.

Purge
The task is terminated. Task termination occurs only when system and data integrity can be maintained.

For all options, if the task is associated with an open TCB in any way (for example, it involves a Java program running on a J8 or J9 mode TCB), you might experience a short delay before the task is finally purged.

Sample screen

The following figure shows the output that is generated after CEKL has been invoked to initially purge a task, for example `CEKL SET TASK(62) PURGE`.

If you attempt to reissue the purge command, the system returns the message informing you that the task is already being purged and it ignores your second command. If you consider that the purge has already taken too long, you can issue a forcepurge command, for example, `CEKL SET TASK(62) FORCEPURGE`.

If the task still remains in the system after an unacceptable length of time, you can issue the kill command, for example, `CEKL SET TASK(62) KILL`. This command always removes the task from your system. You can confirm this by inquiring about the task, for example by entering `CEKL INQ TASK(62)`.

```
F MEACO22X,CEKL SET TA(62) PUR
+IYK2ZFX3 DFHCQ0252I  CEKL SET: PURGE request issued for task number 00062.
+IYK2ZFX3 TAS(00062) PUR SUS QR CEMT               USE(CICSUSER) ATT(00270) CPU(*****) RUA(00005)
+IYK2ZFX3 HTI(00271) HTY(ZCIOWAIT) HVA(DFHZARQ1)
F MEACO22X,CEKL SET TA(62) PUR
+IYK2ZFX3 DFHCQ0253I  CEKL SET: PURGE request ignored; task number 00062 is being purged
+IYK2ZFX3 TAS(00062) PUR SUS QR CEMT               USE(CICSUSER) ATT(00288) CPU(*****) RUA(00005)
+IYK2ZFX3 HTI(00289) HTY(ZCIOWAIT) HVA(DFHZARQ1)
+IYK2ZFX3 DFHCQ0252I  CEKL SET: FORCEPURGE request issued for task number 00062.
+IYK2ZFX3 TAS(00062) FOR SUS QR CEMT               USE(CICSUSER) ATT(00270) CPU(*****) RUA(00005)
+IYK2ZFX3 HTI(00271) HTY(ZCIOWAIT) HVA(DFHZARQ1)
+IYK2ZFX3 DFHCQ0257I  CEKL SET: FORCEPURGE request ignored; task number 00062 is being forcepurged
+IYK2ZFX3 TAS(00062) FOR SUS QR CEMT               USE(CICSUSER) ATT(00288) CPU(*****) RUA(00005)
+IYK2ZFX3 HTI(00289) HTY(ZCIOWAIT) HVA(DFHZARQ1)
F MEACO22X,CEKL SET TA(62) KILL
+IYK2ZFX3 DFHCQ0259I  CEKL SET: KILL request issued for task number 00062.
+IYK2ZFX3 TAS(00062) KIL SUS QR CEMT               USE(CICSUSER) ATT(00394) CPU(*****) RUA(00005)
+IYK2ZFX3 HTI(00395) HTY(ZCIOWAIT) HVA(DFHZARQ1)
F MEACO22X,CEKL I TA(62)
+IYK2ZFX3 DFHCQ0242I  CEKL INQUIRE: no tasks matched selection criteria.
```

Figure 32. CEKL SET TASK output

Chapter 14. CEMN - CICS monitoring facility

Use the CEMN monitoring facility transaction to inquire on the settings for the CICS monitoring facility and to change some of the settings without needing to restart CICS.

CEMN gives you an alternative to the **INQUIRE MONITOR** and **SET MONITOR** system programming commands and the equivalent CEMT commands.

You can change these settings using CEMN:

Monitoring status
Whether monitoring is required.

Exception class
Whether exception class data is required.

Identity class
Whether identity class data is required.

Performance class
Whether performance class data is required.

Resource class
Whether resource class data is required.

DPL resource limit
The maximum number of distributed program link requests for which resource class data is collected.

File resource limit
The maximum number of files for which resource class data is collected.

Tsqueue resource limit
The maximum number of temporary storage queues for which resource class data is collected.

Compression status
Whether monitoring data is to be compressed. The default is for monitoring records to be compressed.

Converse status
Whether separate performance class records are produced for conversational tasks.

Syncpoint status
Whether separate performance class records are produced for sync point requests.

Frequency
The interval at which CICS produces performance class records for long-running tasks. CICS can produce a performance class monitoring record in this way only when the long-running transaction is running on the quasi-reentrant (QR) or concurrent (CO) mode TCBs.

Changing these settings affects the monitoring data that is recorded for tasks that are running at the time you make the change. The effects are the same as if you had changed the settings using the **SET MONITOR** system programming command or the CEMT SET MONITOR command. “CEMT SET MONITOR” on page 484

© Copyright IBM Corp. 1982, 2011

explains how data for running tasks is accumulated, recorded, or lost when you change the settings for the CICS monitoring facility.

These settings are displayed but cannot be changed using CEMN:

Application naming status
Whether CICS application naming support is enabled.

RMI status
Whether additional performance monitoring is active for CICS resource managers.

Time option
Whether the time stamp fields are returned in GMT or local time.

You can change the distributed program link requests limit, file resource limit, temporary storage queue resource limit, application naming status, and RMI status using the DFHMCT TYPE=INITIAL macro in the monitoring control table (MCT). You can change the time option using the MNTIME system initialization parameter. A CICS restart is required to implement any of those changes.

Start the CEMN transaction by typing `CEMN` on the command line of your display and pressing Enter. You get a display of the primary CEMN panel, showing the current state of the CICS monitoring facility and the settings of the monitoring options in your own system. To see the second options panel, press F5.

You can overtype the input fields with the new values that you require. The `Application Naming Status`, `RMI Status`, and `Time Option fields`, are inquire-only fields. When you press Enter, CEMN issues the necessary commands to set the new values. If any errors occur, press F9 to display the error messages. If only one short error message occurs, it is shown near the bottom of the display.

CEMN is a Category 2 transaction.

Chapter 15. CEMT - master terminal

Use CEMT to invoke all the master terminal functions. The master terminal program provides dynamic user control of the CICS system. By using this function, an operator can inquire about and change the values of parameters used by CICS, alter the status of the system resources, terminate tasks, and shut down the CICS system.

The following transactions are subsets of CEMT (that is, they contain some but not all of the functions of CEMT):
- CEBT for alternate CICS systems (see Chapter 9, "CEBT - master terminal (alternate CICS)," on page 77)
- CEOT (see Chapter 16, "CEOT - terminal status," on page 533)
- CEST (see Chapter 20, "CEST - supervisory terminal," on page 551).

The CECI transaction (see Chapter 10, "CECI - command-level interpreter," on page 89)) also uses the same command interpreter as CEMT.

You can start CEMT from IBM 3270 family display devices or equivalent terminals or from the system console. Starting CEMT from the operating system console is described in "Using the system console" on page 9. The description of CEMT throughout this section assumes, unless stated otherwise, that you are using a 3270 family display device or equivalent terminal.

To start CEMT, you can either enter the identifier CEMT on the command line by itself, or you can add options on the same line, separated by commas or blanks, to form a complete request.

If you start the transaction by entering CEMT by itself and pressing ENTER, you get a display that prompts you for more information, as described in "Starting the CEMT transaction."

Starting the CEMT transaction

You can construct a CEMT request in stages, or you can enter a complete request in one go. The CEMT transaction checks the syntax of requests and diagnoses errors. If your request is syntactically correct, it is processed immediately. If your request cannot be processed because of severe syntax errors, the full syntax of your request is displayed.

About this task

If you put a question mark (?) in front of your request, the syntax of your request is displayed but it is not processed.

If you put a minus sign (-) immediately following CEMT, and are using the master terminal transaction from the system console, CEMT always displays its results in abbreviated format. This is independent of the number of objects returned on the inquiry. Without the use of a minus sign, CEMT commands issued from the system console display the results in expanded format only if a single object is returned.

After you have initiated the CEMT transaction, you need not reenter CEMT on subsequent requests, because the identifier is implied at the beginning of any

© Copyright IBM Corp. 1982, 2011

further request. However, if you are using the system console, you must always start each master terminal request with CEMT.

Procedure

1. Enter `CEMT` on the command line of the screen and press the Enter key. You get the following response:

```
 STATUS:  ENTER ONE OF THE FOLLOWING

Discard
Inquire
Perform
Set
```

Figure 33. CEMT transaction: initial screen

2. This display prompts you to enter one of the keywords shown.
   a. If you type `D` and press ENTER, the list of resources types that you can discard are displayed.
   b. If you type `I` and press ENTER, the list of resources types that you can inquire about are displayed.
   c. If you type `P` and press ENTER, the list of resources types that you can perform an action on are displayed.
   d. If you type `S` and press ENTER, the list of resources types that you can change the status of are displayed.

System parameters

If you press ENTER without selecting one of the CEMT keywords, INQUIRE SYSTEM is assumed and you get a display of the system parameters. Full details are given in "CEMT INQUIRE SYSTEM" on page 310.

Program function (PF) keys

When you start CEMT, the lower part of your display contains a list of function keys. If your terminal does not have such keys, you can get the same effect by placing the cursor under a name in the list, and pressing ENTER.

Here is what the function keys do:

F1 The general HELP key. It also gives you a list of all the function keys and what they do.

F2 Switches the display of character values to hexadecimal format on an expanded panel (see "Expanded panels" on page 118).

F3 Ends this master terminal session by terminating the CEMT transaction. If, however, you use this key when you are modifying your display, say by keying in a new command, or by overtyping an old command, this key does not end your session; it is ignored.

F5 Gives a variables display. You can use this to create values for use in a command.

F7 Scrolls backward half a screen.

F8 Scrolls forward half a screen.

F9 Expands messages. If more than one message has been generated in response to your request, a message telling you how many messages have been generated appears near the bottom of the screen. F9 lets you display all the messages on a separate display. Press ENTER to go back to your original display.

F10 Scrolls backward a full screen.

F11 Scrolls forward a full screen.

Minimum abbreviation of keywords

CEMT accepts as few characters of a keyword as needed to identify it uniquely in the request.

For example, you can enter the keyword TASK as TA or TAS, but you cannot use T, because it can be confused with TCLASS, TDQUEUE, TERMINAL, TRANSACTION, or TRDUMPCODE. You are prompted if the abbreviation you use is not unique. An exception to this rule is SHUTDOWN, for which you must specify at least SHUT.

In the syntax displays on your screen, the minimum permitted abbreviation is shown in uppercase characters, with the remainder shown in lowercase characters.

Minimum abbreviations might change between CICS releases because of the introduction of new commands.

ALL option

Many of the request formats contain the ALL option. For INQUIRE, this is the default. For example, if you inquire about terminals, you receive information about all terminals unless you specify a terminal identifier, a terminal class identifier, or a system identifier.

ALL is not the default on a SET command, however. If you specify the ALL option on a SET command, any changes you request are made to all resources of the specified type that you are authorized to access.

If there are no resources of the type specified on the SET command, a response of NOT FOUND is given.

CLASS option

When operating on terminals, transactions, and programs, the CLASS option can be used to identify the suffix of a TLT, XLT, or PLT.

This limits the scope of the INQUIRE or SET commands to those resources defined in the specified table. Any resources whose definitions have not been installed are displayed as NOT FOUND. For more information, see the *CICS Resource Definition Guide*.

Family of resources (using * and + symbols)

You can use the symbols * and + as part of an identifier to specify a family of resources. * represents any number of characters (including none), and + represents a single character, for example:

A*	all identifiers beginning with A
DATA	all identifiers containing the characters DATA
TERM00+	all 7-character identifiers starting with TERM00.

Use of * and + is restricted to selecting existing resources and cannot be used either to create a new set of resources or to discard resources.

Note: The INQUIRE command and the SET command may use the generic name in a different way. The following two examples illustrate the difference:

```
CEMT INQUIRE FI(*) DSN(ABCD*)
```

lists all the files with data sets with a name beginning with ABCD.

```
CEMT SET FI(*) DSN(ABCD*) CLO
```

associates all your files with data set ABCD* and closes them all.

Thus the generic name has the expected effect only for the resource itself.

List of resource identifiers

You can specify a list of resource identifiers, rather than a single one, separating the items in the list by commas or blanks. Any errors of syntax are reported.

```
CEMT SET TERMINAL(TO1,TO2,TO3,TO4) INSERVICE
```

You cannot use * or + when the identifier is in a list of identifiers.

Note: If you use the comma (,) in the name of a resource identifier, you will not be able to use any commands that permit the use of lists of resources, because the comma will be treated as a list separator, and not as a part of a name.

Subdefining resource groups

About this task

When you make an inquiry, you can subdefine a resource group by specifying one or more attributes of the group. For example:

```
CEMT INQUIRE TERMINAL ALL INSERVICE
```

displays the status of all terminals that are in service.

If the subdefined resource group has no members, CEMT returns a NOT FOUND response. For example, if all the files in the system are closed, the command:

```
CEMT INQUIRE FILE ALL OPEN
```

gives the NOT FOUND response.

You cannot subdefine a resource group on the SET command in the same way. For example:

```
CEMT SET TERMINAL ALL INSERVICE PAGE
```

puts all terminals both in service **and** into page status.

However, you can manage such groups of resources by issuing an INQUIRE request, and overtyping the display produced.

Overtyping a display

About this task

When you issue an INQUIRE request, you usually get a display that consists of status information for each resource in the specified group. The status information is displayed as a list of abbreviated keywords. You can move the cursor to some fields and change their contents by overtyping. When you press the ENTER key again, CICS reads the contents of all fields that have been changed, and processes any valid operations implied by the changes. If you make an invalid change, you get an error message, and the field is not changed.

When a field is overtyped, CEMT will accept and identify only as few characters as are necessary to uniquely identify the keyword within the valid options for that position. This is usually only one character.

On many of the status displays, the current status is displayed in certain positional fields at the right-hand side of the display. For some fields, the positional display appears only when the status of a particular resource is ‘on'. You can overtype the current status (when it is either displayed or blank) to change it.

So, for example, if you type ‘CEMT INQUIRE SYD', you get a display of the current settings of the entries in the system dump table. The various attributes of those entries are displayed in fields across the screen. If CICS is to be shut down after the occurrence of an error for this dump code, ‘Shu' appears in the third field of attributes for that dump code. If CICS is not to be shut down, however, the ‘Shu' space is left blank. If you want CICS to be shut down for this dump code, you can tab to that field and type ‘shu' in that blank space in that field only.

The sequence of the positional fields is given in the descriptions of the various CEMT INQUIRE commands.

Whenever you overtype a display, not only is that particular action taken but all the status information is refreshed. You can avoid the overhead of a large number of table searches either by using the SET command, or by limiting your inquiry to a specific number of resources (either by using a ‘wild card' in the search argument or by naming a range of resource identifiers).

Tab key

The fields you can change are different in each display. You can detect them, however, by pressing the tab key repeatedly. This causes the cursor to jump from one field to the next.

Scrolling symbol (+ sign)

A plus (+) sign on the first or last line of a display tells you that there is more data above or below your current display. Scrolling backward reveals data above, and scrolling forward reveals data below.

Blank fields in a display

Some displays contain blank fields that allow you to specify options, such as NEWCOPY, on the CEMT SET PROGRAM command, which are not part of the

status of a resource. A status, such as NOREAD, which begins NO, is also represented by a blank field. You can detect these fields by using the tab key.

Using the ? key

If you precede the command with ?, the syntax for that command appears on the screen.

About this task

For example ?INQUIRE TERMINAL gives the list of options that you can specify with the inquiry command with the default highlighted. See Figure 34.

```
?INQ TERM
 STATUS:  COMMAND SYNTAX CHECK
  CEMT  Inquire TErminal()
   < CLass() | ALl >
   < TRansaction() >
   < PRiority() >
   < PAgeable | AUtopageable >
   < Inservice | Outservice >
   < ATi | NOAti >
   < TTi | NOTti >
   < NEtname() >
   < ACquired | RELeased >
   < CReate | NOCreate >
   < REMotesystem() >
```

Figure 34. Sample of screen showing the syntax of a INQUIRE command

Expanded panels

You can select one resource item from a list, then check and change its options if required.

Type the required CEMT INQUIRE or SET command name, then move the cursor to the relevant line on the display panel and press ENTER. This will result in an expanded display showing the attributes of the selected resource item. In the process, an INQUIRE command is executed for the associated resource.

Move the cursor to the left of the first entry and press ENTER. This selects the first entry and displays the relevant information for that resource; for example, the expanded information for a terminal gives the display in Figure 35 on page 119.

```
in term
RESULT - OVERTYPE TO MODIFY
  Terminal(bell)
  Transaction()
  Priority( 000 )
  Pagestatus( Pageable )
  Servstatus( Inservice )
  Atistatus( Noati )
  Ttistatus( Tti )
  Nature(Local)
  Purgetype(              )
  Netname()
  Termstatus(              )
  Createsess(              )
  Task(00000)
  Remotesystem()
  Rname()
  Rnetname()
  Cid()
```

Figure 35. Terminal settings for a selected terminal

Overtype the displayed settings if you want to make changes. Press ENTER to restore the original panel.

You can view a number of expanded panels in succession by entering '?' or 's' in the space at the beginning of the line for each resource you want to query. After the first expanded panel, pressing ENTER takes you to the next panel.

Hexadecimal values

Some values such as TSQUEUE and NETUOWID may contain nondisplayable characters, which are shown as periods. You can use PF2 on the expanded panel to see the hexadecimal values of these characters. To switch back to character format, press PF2 again.

Variables

PF5 gives a display of variables, similar to that for the CECI command. Here are some examples of its use.

From a FILE display, you might want to inquire of a specific DSNAME, which is typically 26 characters and can be up to 44 characters in length. Go to the expanded panel, position the cursor on the DSNAME line, and press PF5. You now have a variable &DSN with the required value. Type I DSN(&DSN) on the command line, and press ENTER.

You might want to find all the TSQUEUE names that begin with a nondisplayable character; for example, X'FF'. Press PF5 for the variables display. Enter X and 2 to make a variable &X of length 2. Press PF2 to switch to hexadecimal, and overtype '4040' with 'FF5C' ('5C' is hexadecimal for the asterisk (*) symbol). Type I TS(&X) on the command line and press ENTER.

For more information on the variables display, see the *CICS Application Programming Guide*.

Invoking CEMT from an application program

About this task

Many of the **CEMT INQUIRE** and SET operations are available through the **EXEC CICS INQUIRE** and **EXEC CICS SET** commands, which you can invoke from an application program; for programming information, see the *CICS System Programming Reference*.

Security checking

The scope of a particular DISCARD, INQUIRE, SET, or PERFORM command can be limited by a combination of several levels of security checking.

About this task

The normal transaction-attach security is described in the *CICS RACF Security Guide*. CEMT calls the exec interface programs to process the user's commands, in much the same way as CECI or a user's command-level program would. The resource security and the command security used by the command-level programming interface are thus applicable to the master terminal user.

For programming information about command security checking, see the *CICS System Programming Reference*.

CEMT DISCARD commands

The DISCARD command allows you to remove an installed resource definition and its corresponding catalog entry from an active CICS system.

For the following resources, the CICS Explorer Operations views provides a functional equivalent:

Command	Explorer operations view
DISCARD ATOMSERVICE	Atom services view
DISCARD BUNDLE	Bundles view
DISCARD CONNECTION	ISC/MRO connections view
DISCARD DB2CONN	DB2 connections view
DISCARD DB2ENTRY	DB2 entries view
DISCARD DB2TRAN	DB2 transactions view
DISCARD DOCTEMPLATE	Document templates view
DISCARD FILE	Files view
DISCARD IPCONN	IPIC connections view
DISCARD JVMSERVER	JVM servers view
DISCARD LIBRARY	Libraries view
DISCARD MQCONN	Websphere MQ Connections view
DISCARD PIPELINE	Pipelines view
DISCARD PROCESSTYPE	Processtypes view
DISCARD PROGRAM	Programs view
DISCARD TCLASS	Transaction classes view
DISCARD TCPIPSERVICE	TCP/IP services view

Command	Explorer operations view
DISCARD TDQUEUE	TD Queues view
DISCARD TERMINAL	Terminals view
DISCARD TRANSACTION	Transactions view
DISCARD TSMODEL	TS models view
DISCARD URIMAP	URI maps view
DISCARD WEBSERVICE	Web services view

DISCARD does not affect the CICS system definition (CSD) file, and you can therefore reinstate the resource using **CEDA INSTALL**(*resourcename*). DISCARD commands require ALTER access to the resource to provide similar security to that for SET commands (which use UPDATE access). You cannot discard resources that are currently in use or are CICS owned resources (beginning with "DFH" or "C").

For a list of the resource types that you can discard, see "CEMT DISCARD" on page 122.

Programming information about FEPI commands is in the *CICS Front End Programming Interface User's Guide*.

There are two ways to discard a resource:

1. If you know the name of the resource that you want to discard, type:
   ```
   CEMT DISCARD resourcetype(resourcename)
   ```
2. If you do not know the name of the resource, type:
   ```
   CEMT INQUIRE resourcetype
   ```

   and a screen similar to Figure 36 appears. Use the carriage return key to move the cursor to the beginning of the line for the resource or resources you want to discard, and type "D" as shown and then press ENTER.

```
 IN PROG
 STATUS:  RESULTS - OVERTYPE TO MODIFY
D Prog(ABCDTASK) Len(0002936) Ass Pro Ena Pri    Ced
     Res(000) Use(000000)
  Prog(DFHACP  ) Len(0008232) Ass Pro Ena Pri
     Res(001) Use(000002) Any Cex Ful
  Prog(DFHAKP  ) Len(0005672) Ass Pro Ena Pri
     Res(000) Use(000001) Bel Cex Ful
  Prog(DFHAMP  ) Len(0108432) Ass Pro Ena Pri
     Res(000) Use(000009) Any Cex Ful
  Prog(DFHAPATT) Len(0000632) Ass Pro Ena Sha
     Res(005) Use(000018) Any Cex Ful
  Prog(DFHBRCP ) Len(0000000) Ass Pro Ena Pri
     Res(000) Use(000000) Bel Cex Ful
  Prog(DFHCCNV ) Len(0000000) Ass Pro Ena Pri
     Res(000) Use(000000) Any Cex Ful
D Prog(ERRTASK ) Len(0000448) Ass Pro Ena Pri
     Res(000) Use(000000)
```

Figure 36. Example of screen showing the discarding of resources

Note:

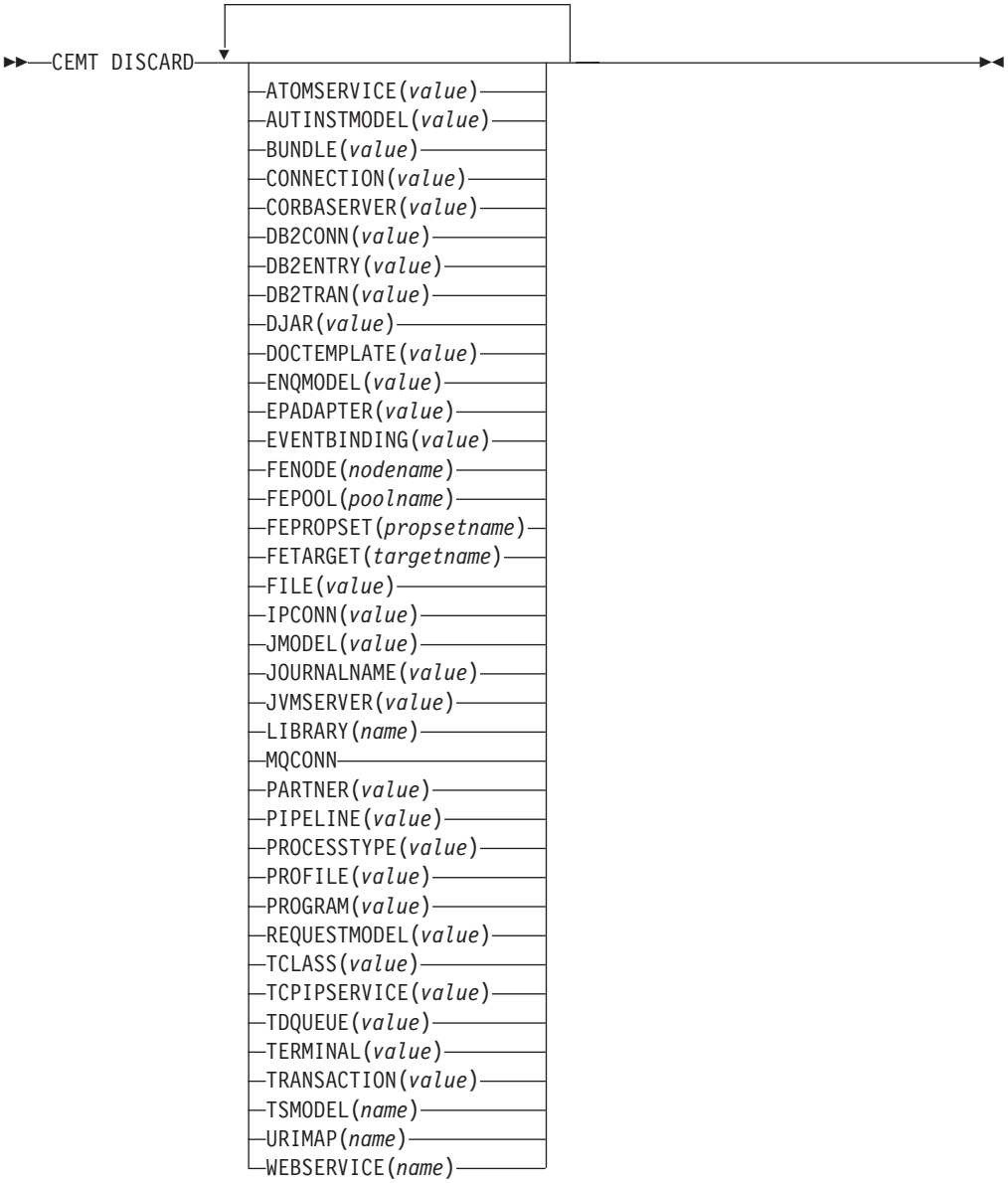
1. A **CEMT INQUIRE** display can be used to discard more than one entry at a time. If you overtype another value as well as typing 'D', the 'D' is ignored.

2. When discarding a resource, the use of * or + in the resource name is not allowed.

CEMT DISCARD

Syntax

CEMT DISCARD



Options

ATOMSERVICE(*value***)**
Specifies the name of the ATOMSERVICE definition to be discarded. The name can be up to 8 characters long. You must disable the ATOMSERVICE resource before discarding it.

AUTINSTMODEL(*value***)**
Specifies the name of the autoinstall model that is to be removed. The name

can be up to 8 characters long. The **DISCARD AUTINSTMODEL** command removes the installed name of an autoinstall model from the CICS system and the CICS catalog.

BUNDLE(*value***)**
Specifies the name of the BUNDLE definition to be discarded. The name can be up to 8 characters long. You must disable the BUNDLE resource before discarding it. Discarding the bundle automatically discards any other resources that were created when the bundle was installed.

CONNECTION(*value***)**
Specifies the 4-character identifier of the CONNECTION definition to be discarded. When a connection is removed, all its associated sessions are also removed.

For deletion to be successful:

- A non-remote connection must be out of service.
- A remote connection must have no active sessions.
- For MRO connections, the interregion communications facility (IRC) must be closed. (You can use **CEMT SET IRC CLOSED** to close it.)
- If the local system is a member of a z/OS Communications Server generic resource group, APPC synclevel 2 connections must not have any outstanding recovery information.

  Other types of connection *can* be discarded, even if there is recovery work outstanding for them. However, it is recommended that you do not discard them if there is. You can use the **INQUIRE CONNECTION RECOVSTATUS** command to check.

In some unusual circumstances, it is possible for deletion of an LU61 connection to fail, even though the connection is out of service, because its sessions are still in-service. If this happens, set the connection INSERVICE, then OUTSERVICE, then reissue the DISCARD command.

CORBASERVER(*value***)**
Specifies the 4-character name of the CORBASERVER definition to be discarded.

If any of the beans installed in the named CorbaServer are in use when the DISCARD command executes, the state of the CorbaServer turns to DISCARDING. No new work is accepted. When all outstanding work has completed, the CORBASERVER definition is removed.

Copies of deployed JAR files are deleted from the CorbaServer's shelf, together with any other files on the shelf that belong to the CorbaServer. The CorbaServer's shelf directory is deleted.

DB2CONN(*value***)**
Specifies that the currently installed DB2CONN is to be removed. You can only discard a DB2CONN when the CICS DB2® interface is not active. Note that discarding a DB2CONN also discards all currently installed DB2ENTRYs and DB2TRANs.

DB2ENTRY(*value***)**
Specifies the name of an installed DB2ENTRY to be removed. A DB2ENTRY can only be discarded when it is in a disabled state.

DB2TRAN(*value***)**
Specifies the name of the DB2TRAN to be removed. Transaction IDs referenced by this DB2TRAN will be routed to the pool after the DB2TRAN is removed.

DJAR(*value***)**
Specifies the 1–32 character name of the DJAR definition to be discarded.

As well as the DJAR definition, the shelf copy of the deployed JAR file and the beans it contains are also removed from the system. The current set of JVM instances are marked for termination, so that each JVM instance will not be reused after it terminates. New JVM instances load all their classes and do not use old versions of classes whose bytecodes have been replaced.

Passivated stateful session bean instances are not deleted by this operation because, in a workload-managed CICS EJB server, they could be activated by other AORs.

References to the home interfaces of enterprise beans in the deployed JAR file are not retracted from the namespace because they might still be installed in other AORs. However, before discarding a deployed JAR file from the last AOR in which it is installed, you might want to issue a **PERFORM DJAR RETRACT** command to retract references to home interfaces.

DOCTEMPLATE(*value***)**
Specifies the name of the document template to be removed.

ENQMODEL(*value***)**
sSpecifies the 8-character identifier of the resource to be discarded.

EPADAPTER(*value***)**
Specifies the name of the EP adapter that is to be removed. You must disable the EP adapter before discarding it.

EVENTBINDING(*value***)**
Specifies the name of the event binding that is to be removed. You must disable the event binding before discarding it.

FENode(*nodename***)**
The name of the FEPI node to be discarded.

FEPOol(*poolname***)**
The name of the FEPI pool to be discarded.

FEPOol(*poolname***)**
The name of the FEPI pool to be discarded.

FEPRopset(*propsetname***)**
The name of the FEPI property set to be discarded.

FETarget(*targetname***)**
The name of the FEPI target to be discarded.

FILE(*value***)**
Specifies the name of the file that is to be removed. The name can be up to 8 characters long. The **DISCARD FILE** command removes the installed name of a file from the CICS system and the CICS catalog.

IPCONN(*value***)**
Specifies the name of the IPCONN to be removed. The name can be up to 8 characters long. You cannot discard an IPCONN unless it is in OUTSERVICE status.

JMODEL(*value***)**
Specifies the name of the installed journal model that you want to remove. The name can be up to 8 characters long. The **DISCARD JMODEL** command removes the journal model name from the CICS system and the CICS catalog. It does not affect any existing journals that have been defined using the journal model

name. These continue to use their existing definitions unless they are discarded using the **DISCARD JOURNALNAME** command.

JOURNALNAME(*value***)**
Specifies the journal name that you want to remove. The name can be up to 8 characters long. The **DISCARD JOURNALNAME** command removes the journal name from the CICS system and the CICS catalog. It does not affect the VSAM data sets currently using the journal as a forward recovery log or autojournal; they continue to use the existing definition until the data sets are closed.

To discard journals defined with a numeric identifier for auto-journaling or forward recovery logs, specify the journal name as DFHJ*nn*, where *nn* is the journal number in the range 1–99.

Note: Specifying DFHJ01 on this command refers to a user journal, *DFHJ01*.

Any future reference to the journal name will cause it to be reinstalled using the journal model resource in effect at that time.

JVMSERVER(*value***)**
Specifies the name of a JVMSERVER resource that you want to remove. The name can be up to 8 characters long. The **DISCARD JVMSERVER** command removes the JVMSERVER definition from the CICS system and the CICS catalog.

LIBRARY(*name***)**
Specifies the name of a LIBRARY resource that you want to remove. The name can be up to eight characters long. The LIBRARY must be disabled. The **DISCARD LIBRARY** command removes an installed resource definition and its corresponding catalog entry from an active CICS system.

Specifying a LIBRARY name of DFHRPL is invalid and results in the message `NOT VALID FOR RPL` being displayed.

MQCONN
Specifies that the currently installed MQCONN definition is to be removed. You can discard an MQCONN definition when the connection between CICS and WebSphere MQ is not active. When you discard an MQCONN definition, the associated MQINI definition is also discarded.

PARTNER(*value***)**
Specifies the name of the partner that is to be removed. The name can be up to 8 characters long. The **DISCARD PARTNER** command removes the installed name of a partner from the CICS system and the CICS catalog.

PIPELINE(*value***)**
Specifies the name of the PIPELINE that is to be removed. The name can be up to 8 characters long. The **DISCARD PIPELINE** command removes the PIPELINE from the CICS system and the CICS catalog.

PROCESSTYPE(*value***)**
Specifies the 8–character name of the PROCESSTYPE to be removed. Only disabled processtypes can be discarded. If you are using BTS in a sysplex, it is strongly recommended that you use CPSM to remove them.

PROFILE(*value***)**
Specifies the name of the profile that is to be removed. The name can be up to 8 characters long. The **DISCARD PROFILE** command removes the installed name of a profile from the CICS system and the CICS catalog.

PROGRAM(*value***)**
Specifies the name of the program, map set, or partition set that is to be

removed. The name can be up to 8 characters long. The DISCARD PROGRAM command removes the installed name of a program, map set, or partition set from the CICS system and the CICS catalog.

REQUESTMODEL(*value***)**
Specifies the name of the REQUESTMODEL resource definition to be removed.

TCLASS(*value***)**
Specifies the name of the transaction class that is to be removed. The name can be up to 8 characters long. The DISCARD TCLASS command removes the installed name of a transaction class from the CICS system and the CICS catalog.

Note: 'TRANCLASS' is also accepted as a synonym for TCLASS.

TCPIPSERVICE(*value***)**
Specifies the name of the TCPIP service to be removed.

TDQUEUE(*value***)**
Specifies the name of the transient data queue that is to be removed from a running CICS system. The name can be up to 4 characters long.

Note: When you discard a transient data queue, an implicit DELETEQ command is issued to free the control intervals used by the queue if the queue is an intrapartition queue. If the queue is a logically recoverable intrapartition queue, a syncpoint is taken after the DELETEQ request, but before the queue definition is discarded. This ensures that the associated control intervals have been freed before the definition is discarded.

TERMINAL(*value***)**
Specifies the name of the terminal to be removed. The name can be up to 4 characters long. The **DISCARD TERMINAL** command removes the installed name of a terminal from the CICS system and the CICS catalog.

Only terminals that use z/OS Communications Server as an access method or consoles are eligible for deletion. Terminals that represent sessions cannot be deleted with **DISCARD TERMINAL**; use the **DISCARD CONNECTION** command instead.

TRANSACTION(*value***)**
Specifies the name of the transaction that is to be removed. The name can be up to 8 characters long. The **DISCARD TRANSACTION** command removes the installed name of a transaction from the CICS system and the CICS catalog.

TSMODEL(*name***)**
Specifies the name of the TS model that is to be removed. The name can be up to 8 characters long. You cannot discard a TSmodel that is in use, or one which has a name beginning with DFH.

URIMAP(*name***)**
Specifies the name of the URIMAP definition that is to be removed. The name can be up to 8 characters long. The **DISCARD URIMAP** command removes the URIMAP definition from the CICS system and the CICS catalog. For a URIMAP definition to be removed, it must be disabled individually (using the **SET URIMAP** command). Disabling a virtual host (using the SET HOST command) does not allow the URIMAP definitions that make up the virtual host to be removed.

WEBSERVICE(*name***)**
Specifies the name of the WEBSERVICE that is to be removed. The name can be up to 8 characters long. The **DISCARD WEBSERVICE** command removes the WEBSERVICE from the CICS system and the CICS catalog.

CEMT INQUIRE commands

Most CEMT requests either inquire about (INQUIRE), or change (SET) the status of one or more named instances of a resource (such as a terminal), a particular subgroup of resources (such as a class of terminals), or all resources of a particular type (such as all terminals).

The INQUIRE command causes the status of the specified resources to be displayed. The SET command makes the changes that you specify, and displays the new status of the resources. No changes are made if there are syntax errors in the SET command.

If, for example, you want to inquire about a resource, enter INQUIRE (or a suitable abbreviation) on the command line. The keywords you can use with CEMT INQUIRE are described in the rest of this section.

If you want to perform those functions that are not concerned with resource status, enter PERFORM (or a suitable abbreviation) on the command line. The keywords you can use with PERFORM are described in “CEMT PERFORM commands” on page 408.

And finally, if you want to change the attributes of a resource, enter SET (or a suitable abbreviation) on the command line. The keywords you can use with SET are described in “CEMT SET commands” on page 425.

You can inquire about any of the options by typing its keyword after INQUIRE on the command line. For example,

```
INQUIRE PROGRAM
```

gives you the status of all programs, and for each program gives its attributes. Full details are given in “CEMT INQUIRE PROGRAM” on page 288.

CEMT INQUIRE ATOMSERVICE

Retrieve information about ATOMSERVICE resource definitions.

In the CICS Explorer, the Atom services operations view provides a functional equivalent to this command.

The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type CEMT INQUIRE ATOMSERVICE. You obtain a display that lists the current status.
- Type CEMT INQUIRE ATOMSERVICE followed by the attributes that are necessary to limit the range of information that you require. So, for example, if you enter CEMT INQUIRE ATOM FEED, the resulting display shows you the details of only those ATOMSERVICE resource definitions that are for Atom feed documents.

To enable or disable the ATOMSERVICE resource definition:

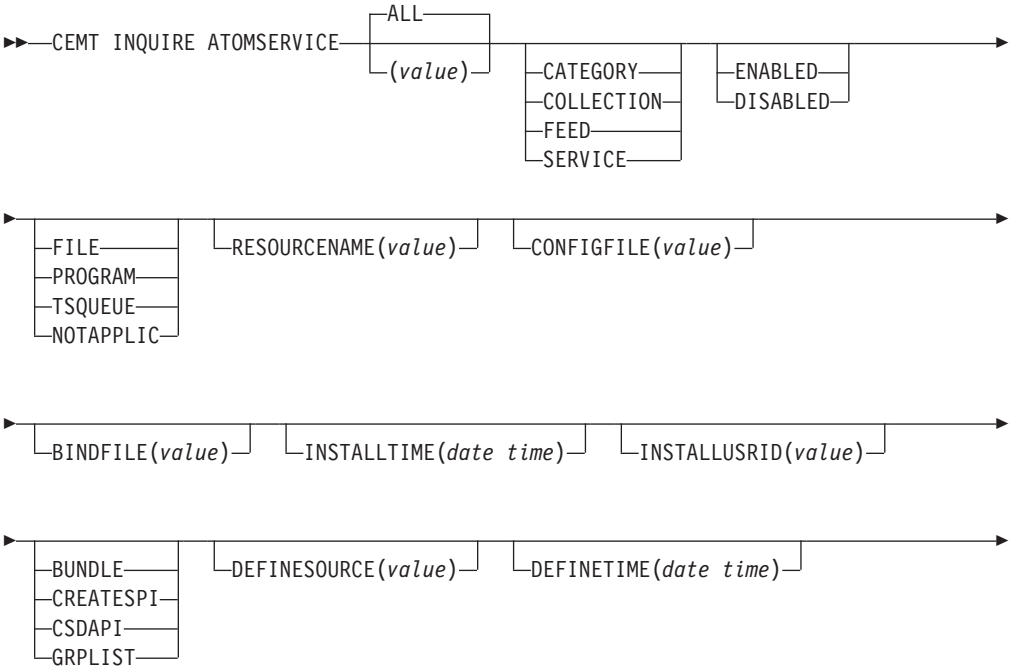
- Enable or disable the BUNDLE resource if the Atom feed is deployed in a bundle. The resource signature of the ATOMSERVICE resource indicates how it was created.
- Overtype your changes on the INQUIRE screen after tabbing to the appropriate field.
- Use the **CEMT SET ATOMSERVICE** command.

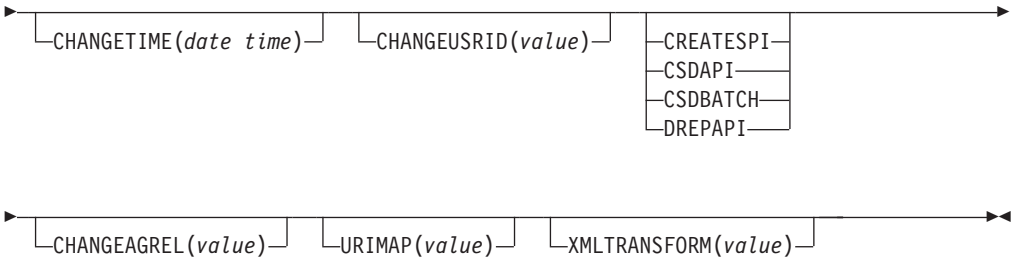
ALL
Is the default. Information about all ATOMSERVICE definitions is displayed, unless you specify a selection of definitions to be queried.

(value)
Is the name (1 - 8 characters) of an installed ATOMSERVICE definition.

If you place the cursor against a specific entry in the list and press ENTER, CICS displays an expanded format of all the attributes for the specified ATOMSERVICE definition.

CEMT INQUIRE ATOMSERVICE





Displayed fields

ATOMTYPE(*value*)
Displays the type of Atom document that is produced by this ATOMSERVICE definition. The possible values are as follows:

CATEGORY
An Atom category document, which lists the categories of documents in a collection.

COLLECTION
An Atom collection document, which contains a group of entry documents that can be edited.

FEED
An Atom feed document, which describes the metadata for a feed, and contains entry documents that provide data for the feed.

SERVICE
An Atom service document, which provides information about the collections of entry documents that are available on the server and can be added to or edited.

BINDFILE(*value*)
Displays the name and file path of the XML binding specified in this ATOMSERVICE resource definition. The XML binding is stored in z/OS UNIX System Services, and it specifies the data structures used by the CICS resource that supplies the data for the Atom document that is returned for this resource definition. Service and category documents do not use an XML binding; if the value of ATOMTYPE is SERVICE or CATEGORY, this field is blank.

CHANGEAGENT(*value*)
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

CHANGEAGREL(*value*)
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time*)
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value*)
Displays the 8-character user ID that ran the change agent.

CONFIGFILE(*value*)
Displays the name and file path of the Atom configuration file specified in this ATOMSERVICE resource definition. The Atom configuration file is stored in z/OS UNIX System Services, and it contains XML that specifies metadata and content for the Atom document.

DEFINESOURCE(*value*)
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time*)
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

ENABLESTATUS(*value*)
Displays the overall status of the ATOMSERVICE resource definition. The possible values are as follows:

ENABLED
The ATOMSERVICE resource definition is enabled.

DISABLED
The ATOMSERVICE resource definition is disabled. An ATOMSERVICE definition with this status can be discarded.

INSTALLAGENT(*value*)
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

BUNDLE
The resource was installed by a bundle deployment.

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

INSTALLTIME(*date time*)
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value*)
Displays the 8-character user ID that installed the resource.

RESOURCENAME(*value*)
Displays the name of the CICS resource that provides the data for this Atom feed or collection. This field does not apply for an Atom service or category document.

RESOURCETYPE(*value*)
Displays the type of CICS resource that provides the data for this Atom feed or collection. This field does not apply for an Atom service or category document. The possible values are as follows:

FILE
A CICS file.

PROGRAM
A service routine, which is a CICS application program written to supply content for Atom entries.

TSQUEUE
A temporary storage queue.

NOTAPPLIC
If the value of ATOMTYPE is SERVICE or CATEGORY, the resource type is not applicable.

URIMAP
Displays the 8-character URIMAP name that indicates the URI associated with this ATOMSERVICE definition. If there is no dynamically generated URIMAP associated with this ATOMSERVICE definition, this field is empty.

XMLTRANSFORM
Displays the 32-character name of the XMLTRANSFORM resource associated with the ATOMSERVICE definition. If the value of ATOMTYPE is SERVICE or CATEGORY, this field is empty.

CEMT INQUIRE AUTINSTMODEL

Find out whether a named autoinstall model is present.

Description

The **INQUIRE AUTINSTMODEL** command returns the names of all the autoinstall models installed in your system. There is no **CEMT SET AUTINSTMODEL** function, but you can use the **INQUIRE AUTINSTMODEL** screens to use the DISCARD function.

For information about autoinstall models, see the *CICS Resource Definition Guide*.

Input

Press the Clear key and type `CEMT INQUIRE AUTINSTMODEL` (or suitable abbreviations for the keywords).

The resulting display is similar to the following example screen.

If you do not type the name of a specific autoinstall model, the names of all the autoinstall models are displayed. Alternatively, you can type the autoinstall model name, which is up to eight characters long; for example, `cemt i auti(dfhlu0e2)`.

Sample screen

```
 IN AUTIN
 STATUS:  RESULTS
  Auti(DFHLU0E2)
  Auti(DFHLU0M2)
  Auti(DFHLU0M3)
  Auti(DFHLU0M4)
  Auti(DFHLU0M5)
  Auti(DFHLU2  )
  Auti(DFHLU2E2)
  Auti(DFHLU2E3)
  Auti(DFHLU2E4)
  Auti(DFHLU2M2)
  Auti(DFHLU2M3)
  Auti(DFHLU2M4)
  Auti(DFHLU2M5)
  Auti(DFHLU3  )
  Auti(DFHLU62T)
  Auti(DFHSCSP )
  Auti(DFH3270 )
+ Auti(DFH3270P)
```

Figure 37. CEMT INQUIRE AUTINSTMODEL screen

Note: The plus symbol (+) in the left margin by the last item on the screen indicates that further model names are displayed when you press the PF8 key.

CEMT INQUIRE AUTINSMODEL

Options

ALL
Returns the names of all the autoinstall models installed in your system.

CEMT INQUIRE AUTOINSTALL

Retrieve information and parameters that control the automatic installation (autoinstall) of z/OS Communications Server terminals, APPC connections, virtual terminals (bridge facilities) used by the 3270 bridge mechanism, and MVS consoles.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

The **INQUIRE AUTOINSTALL** command displays information about autoinstall values. For information about autoinstall, see the *CICS Resource Definition Guide*.

Input

Press the Clear key and type `CEMT INQUIRE AUTOINSTALL` (or suitable abbreviations for the keywords).

The resulting display is similar to the following example. To change any of the values shown, tab to the highlighted fields and overtype them as required.

Sample screen

```
INQ AUTOINSTALL
STATUS:  RESULTS - OVERTYPE TO MODIFY
 Auto Max( 100 ) Pro( DFHZATDX ) Cur(000) Pro Ena
```

Figure 38. CEMT INQUIRE AUTOINSTALL screen

If you place the cursor against the result and press ENTER, CICS displays an expanded format, as shown in Figure 39.

```
IN AUTOIN
RESULT - OVERTYPE TO MODIFY
  Autoinstall
  Maxreqs( 100 )
  Program( DFHZATDX )
  Curreqs( 000 )
  Consoles( Fullauto )
  Enablestatus( Enabled )
  Aibridge( Autotermid )
```

Figure 39. The expanded display of an individual entry

CEMT INQUIRE AUTOINSTALL

Displayed fields

AIBRIDGE
Indicates whether the autoinstall user replaceable module (URM) is called for bridge facilities. The values are as follows:

AUTOTERMID
Bridge facilities are defined automatically by CICS. The autoinstall URM is not called.

URMTERMID
The autoinstall URM is called.

AUTOINSTALL
Identifies that this panel relates to terminal autoinstall.

CONSOLES(*value*)
Displays the status of autoinstall for MVS consoles in the CICS region. The values are as follows:

FULLAUTO
Autoinstall for consoles is active without any support from the autoinstall control program. CICS generates the required terminal ID automatically without calling the control program, and sets the default delete-delay period (60 minutes) for the console entry.

PROGAUTO
Autoinstall for consoles is active, with the support of the autoinstall control program (specified on the **AIEXIT** system initialization

parameter). The autoinstall control program is required to provide the terminal ID and, optionally, a delete-delay period.

NOAUTO
Autoinstall for consoles is not active.

You can change this attribute by overtyping it with a different value.

CURREQS(*value***)**
Displays the number of autoinstall requests that are currently being processed. This is not the number of terminals that are currently installed using autoinstall.

ENABLESATUS(*value***)**
Displays the overall status of the terminal autoinstall facility, indicating whether CICS is enabled to install z/OS Communications Server terminals and MVS consoles. The values displayed are as follows:

ENABLED
CICS can currently install consoles, terminals, or both. To determine which of these the facility is enabled for, check the values returned on the MAXREQS and CONSOLES options. Enabled is returned for the following conditions:

TERMINALS
MAXREQS not equal 0 and autoinstall control program is enabled.

CONSOLES
1. CONSOLES displays FULLAUTO.
2. CONSOLES displays PROGAUTO and the autoinstall control program is enabled.

DISABLED
Neither consoles nor terminals can be autoinstalled in CICS. DISABLED is returned for the following conditions:

TERMINALS
MAXREQS equal 0, or autoinstall control program is disabled.

CONSOLES
1. CONSOLES displays NOAUTO.
2. CONSOLES displays PROGAUTO but the autoinstall control program is disabled.

MAXIMUM(*value***)**
Displays the largest number of autoinstall requests that are allowed to queue at any one time. This is not the largest number of terminals that can be installed using autoinstall.

Note: You can reset this value by overtyping it with a different value.

PROGRAM(*value***)**
Displays the name of the program that controls the autoinstall process. This is either the CICS-supplied default autoinstall program, DFHZATDX, or a user-written program.

You can reset this value by overtyping it with a different value.

CEMT INQUIRE AUXTRACE

Retrieve information about the status of auxiliary trace.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

Auxiliary trace entries are made to a pair of CICS-controlled BSAM data sets, A and B.

You can use the **INQUIRE AUXTRACE** command to see the following information:

- Whether the auxiliary trace data set is open or closed
- Which auxiliary trace data set is active, Cur(A) or Cur(B)
- Whether an auxiliary trace is in progress (started) or not (stopped).

You can start, stop, or pause tracing, and specify that automatic switching of the auxiliary data set occurs when needed. You can use the SWITCH option to change the data set.

Instead of using the **SET** command, you can overtype your changes on the INQUIRE screen after tabbing to the appropriate field (see "Overtyping a display" on page 117).

For information on using auxiliary tracing, see the *CICS Problem Determination Guide*. See also the CETR transaction, Chapter 21, "CETR - trace control," on page 553.

Input

Press the Clear key and type `CEMT INQUIRE AUXTRACE` (or suitable abbreviations for the keywords).

The resulting display is similar to the following example. To change any of the values shown, tab to the highlighted or blank fields and overtype them as required.

Sample screen

```
I AUX
STATUS:  RESULTS - OVERTYPE TO MODIFY
 Aux Cur(A) Sto
```

Figure 40. CEMT INQUIRE AUXTRACE screen

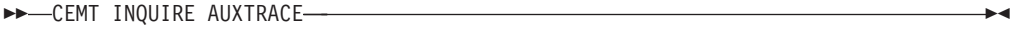
Note: When a value does not apply, or is negative (the value begins with No), the fields on the screen are blank. To modify these fields, locate them by tabbing (they appear in the same sequence as in the expanded format), and overtype with input valid for that field. You might find it more convenient to use the expanded format when setting one of these values.

If you place the cursor against the result and press ENTER, CICS displays an expanded format as, shown in Figure 41 on page 136.

```
I AUX
RESULT - OVERTYPE TO MODIFY
  Auxtrace
  Curauxds(A)
  Status( Stop )
  Switchstatus( Noswitch )
  Switchaction(             )
```

Figure 41. The expanded display of an individual entry

CEMT INQUIRE AUXTRACE



Displayed fields

AUXILIARY
Identifies this panel as relating to auxiliary trace.

CURAUXDS(*value***)**
Displays a 1-character identifier of the current auxiliary data set, which can be A or B. This value is blank if there is no auxiliary trace data set.

STATUS(*value***)**
Displays the status of auxiliary tracing in your CICS system. The values are as follows:

START
Auxiliary tracing is in progress (the auxiliary trace data set is open).

PAUSE
Auxiliary tracing has stopped, but the trace data set has been left open. A subsequent START request causes trace entries to be written immediately following those that were written before the PAUSE request.

STOP
Auxiliary tracing has stopped, and the trace data set is closed.

Note: You can reset this value by overtyping it with a different value.

SWITCHACTION*(value) (input only field)*
Specifies that the trace data set is to be switched. The value is:

SWITCH
Makes an immediate switch from the current auxiliary trace data set to the alternate trace data set.

SWITCHSTATUS(*value***)**
Displays whether automatic data set switching occurs when the current auxiliary trace data set becomes full. The values are as follows:

NOSWITCH
Switching does not occur without operator intervention.

NEXT
Automatic switching occurs when the current trace data set is full, but not subsequently.

ALL Automatic switching occurs as necessary until the end of this CICS session, without the need for operator intervention.

Note: You can reset this value by overtyping it with a different value.

CEMT INQUIRE BEAN

Retrieve information about an installed enterprise bean.

Description

INQUIRE BEAN returns information about any currently installed enterprise bean.

Input

Press the Clear key and type `CEMT INQUIRE BEAN`. You get a display that lists the names and status of any installed enterprise bean. Note that you cannot change any of the displayed information.

Sample screen

```
INQ BEAN
STATUS:  RESULTS
 Bean(Managed                  ) Corba(CORA) Djar(IJAR                          )
 Bean(Mandatory                ) Corba(CORB) Djar(BJAR                          )
 Bean(MasterTs                 ) Corba(CORA) Djar(IJAR                          )
 Bean(Never                    ) Corba(CORB) Djar(BJAR                          )
 Bean(NotSupported             ) Corba(CORA) Djar(IJAR                          )
 Bean(Required                 ) Corba(CORB) Djar(BJAR                          )
 Bean(RequiresNew              ) Corba(CORA) Djar(IJAR                          )
 Bean(Supports                 ) Corba(CORB) Djar(BJAR                          )









  SYSID=QT20 APPLID=IYCQST20
RESPONSE: NORMAL                            TIME:  11.09.09  DATE: 19.01.01
  PF 1 HELP       3 END       5 VAR       7 SBH 8 SFH 9 MSG 10 SB 11 SF
```

Figure 42. CEMT INQUIRE BEAN screen

If you place the cursor against a specific entry in the list and press ENTER, CICS displays an expanded format.

```
i bean
RESULT
  Bean(Managed)
  Corbaserver(CORA)
  Djar(IJAR                          )




SYSID=QT20 APPLID=IYCQST20
TIME:  11.10.42  DATE: 19.01.01
  PF 1 HELP 2 HEX 3 END       5 VAR       7 SBH 8 SFH       10 SB 11 SF
```

Figure 43. The expanded display of an individual entry

If you inquire against a specific bean and press ENTER, CICS displays an expanded format.

```
inq bean(Managed) corba(CORA)
STATUS:  RESULTS
 Bean(Managed                   ) Corba(CORA)
 Djar(IJAR                            )



SYSID=QT20 APPLID=IYCQST20
TIME:  11.10.42  DATE: 19.01.01
  PF 1 HELP 2 HEX 3 END       5 VAR        7 SBH 8 SFH       10 SB 11 SF
```

Figure 44. The expanded display of a specific entry

CEMT INQUIRE BEAN

Displayed fields

ALL
This value is the default.

(*value***)**
Specify a 1 - 240 character bean name.

CORBASERVER (*value***)**
Displays a 4-character CorbaServer name.

DJAR (*value***)**
Displays a 1 - 32 character deployed JAR file name.

CEMT INQUIRE BRFACILITY

Retrieve information about a virtual terminal (bridge facility) used by the 3270 bridge mechanism.

Description

The **INQUIRE BRFACILITY** command returns information about a bridge facility. This is a virtual terminal used by the 3270 bridge mechanism to simulate a real 3270 when running a CICS 3270 application in a bridged environment.

Input

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT INQUIRE BRFACILITY` (or suitable abbreviations for the keywords). The resulting display lists the status of any currently allocated bridge facilities.
- Type `CEMT INQUIRE BRFACILITY` (or suitable abbreviations for the keywords), followed by the attributes that are necessary to limit the range of information that you require. For example, if you enter `CEMT I BR S`, the resulting display shows the details of only those bridge facilities that were allocated for Link3270 requests (SHARED). To find out which options you can enter on the `CEMT I BR` request, you can enter `?CEMT I BR`, for example.

```
?CEMT I BR
 STATUS:  COMMAND SYNTAX CHECK
  CEMT  Inquire BRfacility()
   < ALl >
   < Netname() >
   < TErmid() >
   < TRansaction() >
   < TAskid() >
   < Userid() >
   < Keeptime() >
   < RELeased >
   < ACquired | AVailable >
   < LINKSYSTem() >
   < LINKSYSNet() >
   < REMOTESYSTem() >
   < REMOTESYSNet() >
   < LOcal | Shared >
```

Figure 45. CEMT INQUIRE BRFACILITY syntax screen

You can change various attributes in the following ways:

- Overtype your changes on the INQUIRE screen after tabbing to the appropriate field (see "Overtyping a display" on page 117).
- Use the **CEMT SET BRFACILITY** command.

Sample screen

```
  I BR
 STATUS:  RESULTS - OVERTYPE TO MODIFY
  Brf(0008000100000016) Net(AAA.    ) Ter(AAA.) Tra(NACT)
     Tas(0005730) Use(JANEDJ  ) Kee(97105) Acq

                                                  SYSID=ST50 APPLID=IYCSST50
  RESPONSE: NORMAL                            TIME:  11.13.29  DATE: 08.05.01
PF 1 HELP       3 END       5 VAR        7 SBH 8 SFH 9 MSG 10 SB 11 SF
```

Figure 46. CEMT INQUIRE BRFACILITY screen

If you place the cursor against a specific entry in the list and press ENTER, CICS displays an expanded format, as shown:

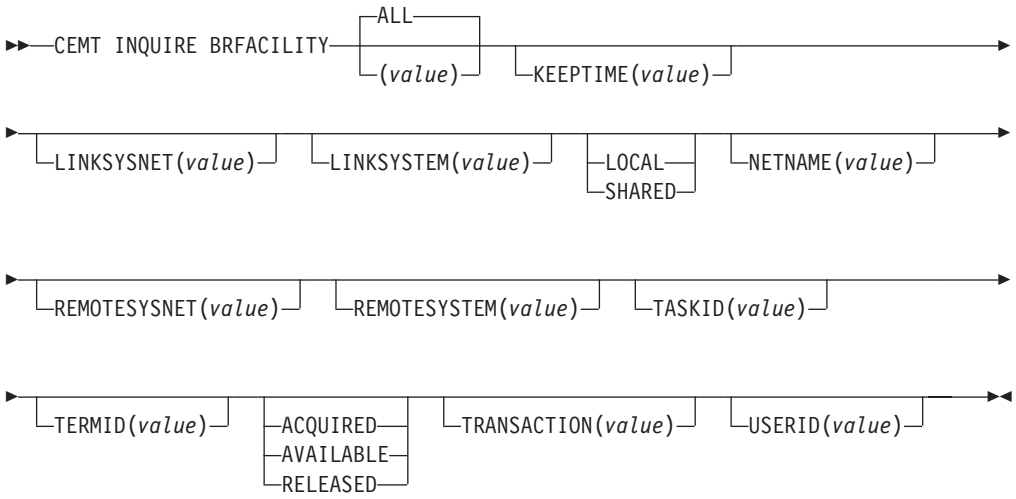
```
I BR
RESULT - OVERTYPE TO MODIFY
  Brfacility(0008000100000016)
  Netname(AAA.)
  Termid(AAA.)
  Transaction(NACT)
  Taskid(0005730)
  Userid(JANEDJ)
  Keeptime(96908)
  Termstatus( Acquired )
  Linksystem()
  Linksysnet()
  Remotesystem(CTGC)
  Remotesysnet( DFHGEN)
  Namespace(Shared)




                                          SYSID=ST50 APPLID=IYCSST50
                                       TIME:  11.16.46  DATE: 08.05.01
PF 1 HELP 2 HEX 3 END       5 VAR       7 SBH 8 SFH       10 SB 11 SF
```

Figure 47. The expanded display of an individual bridge facility entry

CEMT INQUIRE BRFACILITY



Displayed fields

(*value***)**
The 8-byte bridge facility token of a specific bridge facility to be displayed.

ALL
The default. All bridge facilities are displayed.

KEEPTIME *(value)*
Displays a full word binary field that shows the length of time (in seconds) that the bridge facility is kept if it is inactive.

- If the bridge facility being displayed is a Link-Bridge, this value is the keep time specified when the facility was allocated, or a default value of 5 minutes if no keep time was specified. :

If the keep time is larger than the value of SIT parameter **BRMAXKEEPTIME**, it is reduced to BRMAXKEEPTIME.

- If the facility being displayed is a Web-Bridge, the keep time is initially set as the web delay terminal keep time (the second part of the **WEBDELAY** SIT parm).

LINKSYSNET(*value***)**
Displays the 8-byte applid of the AOR if the Link3270 bridge request is routed to another region. If the request is processed in the same region as the Link3270 router, this field is blank. This field might change if dynamic transaction routing makes more than one attempt at running the first transaction in a Link3270 session. This field is only set in the router region.

LINKSYSTEM(*value***)**
Returns the 4-byte sysid of the AOR if the Link3270 bridge request is routed to another region. If the request is processed in the same region as the Link3270 router, this field is blank. This field migh change if dynamic transaction routing makes more than one attempt at running the first transaction in a Link3270 session. This field is only set in the router region.

NAMESPACE
Displays the scope of the namespace used to allocate bridge facility names. Values are as follows:

LOCAL
The bridge facility was allocated by the START BREXIT bridge mechanism, so its name is unique only in the local region where it is created.

SHARED
The bridge facility was allocated by the Link3270 bridge mechanism, so its name is unique across all CICS router regions in the CICSplex who have access to a shared DFHBRNSF namespace file.

NETNAME(*value***)**
Displays the 8-byte virtual netname name of the bridge facility about which you are inquiring.

REMOTESYSNET(*value***)**
Displays an 8-byte field that shows the applid of the bridge router region. This field is only set in the AOR region. It is blank if the request is processed in the router region and not sent to an AOR.

REMOTESYSTEM(*value***)**
Displays a 4-byte field that shows the sysid of the bridge router region. This field is only set in the AOR region. It is blank if the request is processed in the router region and not sent to an AOR.

TASKID(*value***)**
Displays a full word binary field showing the number of the task running the user transaction. This field is only set in the AOR. This field is zero the bridge facility is currently not in use.

TERMID(*value***)**
Displays the 4-byte virtual terminal name of the bridge facility about which you are inquiring.

TERMSTATUS
Displays the status of the bridge facility. Values are as follows:

ACQUIRED
The bridge facility is currently in use.

AVAILABLE
The bridge facility is not in use. It can be reused by the client.

RELEASED
SET BRFACILITY RELEASED has been issued for the bridge facility. It will be deleted on the next cleanup cycle.

TRANSACTION(*value***)**
Displays a 4-byte field that shows the name of the user transaction being run by the 3270 bridge, as known in the current region. This is blank if the bridge facility is currently not in use.

USERID(*value***)**
Displays an 8-byte field that shows the userid associated with this bridge facility.

CEMT INQUIRE BUNDLE

Retrieve information about a BUNDLE resource.

In the CICS Explorer, the Bundles operations view provides a functional equivalent to this command.

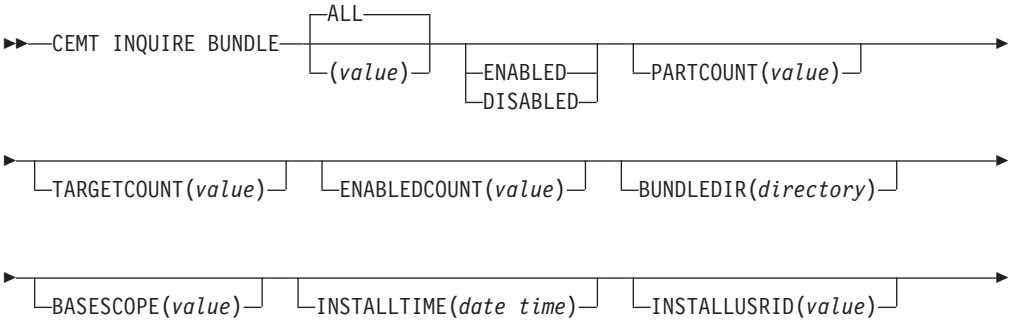
The resource signature

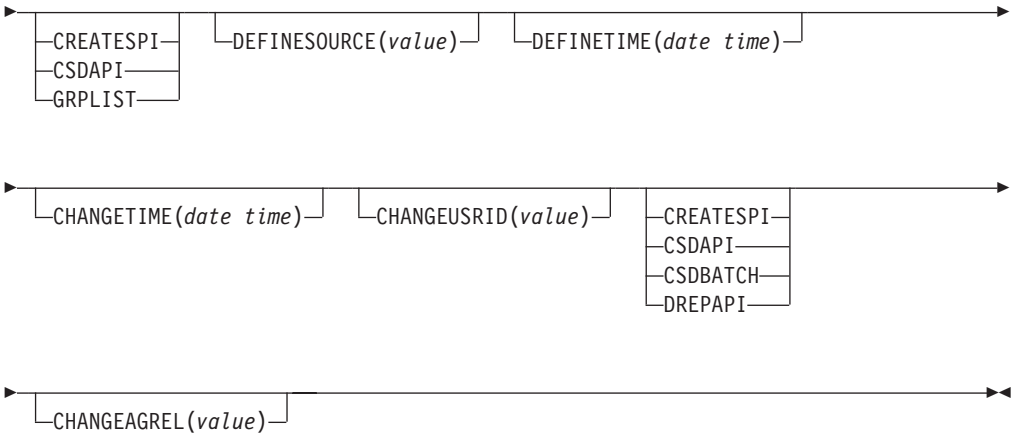
You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key to clear the screen and then enter `CEMT INQUIRE BUNDLE` to obtain information about one or more bundles that are installed in the CICS region.

CEMT INQUIRE BUNDLE





Displayed fields

BASESCOPE(*value***)**
Displays the URI that defines the root namespace for the contents of the bundle.

BUNDLEDIR(*directory***)**
Displays the directory on z/OS UNIX where the bundle is deployed.

CHANGEAGENT(*value***)**
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

DEFINESOURCE(*value***)**
Displays the source of the resource definition. The DEFINESOURCE value

depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

ENABLEDCOUNT(*value***)**
Displays the current number of resources that were dynamically created by the bundle and are enabled in the CICS region.

ENABLESTATUS(*value***)**
Displays the overall status of the bundle. The possible values are as follows:
ENABLED
The bundle is enabled for use.
DISABLED
The bundle is disabled.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

PARTCOUNT(*value***)**
Displays the total number of imports, exports, and definition statements that are defined in the bundle manifest.

TARGETCOUNT(*value***)**
Displays the total number of dynamically created resources in the bundle. CICS automatically enables the BUNDLE resource when all of the dynamically created resources are in an enabled state.

CEMT INQUIRE CFDTPOOL

Retrieves information about coupling facility data table (CFDT) pools.

Description

INQUIRE CFDTPOOL returns information about any currently installed CFDT pool names. CICS installs these automatically whenever it installs a file resource definition that refers to a CFDT.

Input

Press the Clear key to clear the screen, and type CEMT INQUIRE CFDTPOOL (or suitable abbreviations for the keywords). The resulting display lists the names and status of any coupling facility data table pools to which CICS is connected through a CFDT server. You cannot change any of the displayed information.

Sample screen

```
inquire cfdt
STATUS:  RESULTS
 Cfd(DTPOOL1 ) Con









                                        SYSID=HA61 APPLID=CICSHA61
  RESPONSE: NORMAL                     TIME:  15.54.22  DATE: 09.03.98
PF 1 HELP       3 END       5 VAR       7 SBH 8 SFH 9 MSG 10 SB 11 SF
```

Figure 48. CEMT INQUIRE CFDTPOOL screen

CEMT INQUIRE CFDTPOOL

Displayed fields

ALL
This is the default. CICS displays the names of all installed CFDT pools.

CFDTPOOL(*value*)
Displays information about a named CFDT pool.

CICS creates a resource entry for a CFDT pool when a file resource definition that refers to a coupling facility data table is first installed. The resource entry is cataloged, and reinstalled from the catalog on a warm or emergency restart.

Note: You cannot discard CFDT pool entries other than by a cold or initial start of the CICS region.

CONNSTATUS
Displays the connection status for the named CFDT pool. The values are as follows:

CONNECTED
The server for the coupling facility data table pool is available in this MVS image, and CICS is currently connected to it.

UNCONNECTED
The server for the coupling facility data table pool is available in this MVS image, but CICS is not currently connected to it.

UNAVAILABLE
The server for the coupling facility data table pool is currently unavailable in the MVS image.

CEMT INQUIRE CLASSCACHE

Retrieves information about the shared class cache for pooled JVMs.

In the CICS Explorer, the JVM classcache operations view provides a functional equivalent to this command.

Description

You can use the **CEMT INQUIRE CLASSCACHE** command to return information about the shared class cache that is used by pooled JVMs in the CICS region. The command also reports the presence of any old shared class caches that are awaiting deletion. You cannot use this command to return information about class caches that are used by JVM servers.

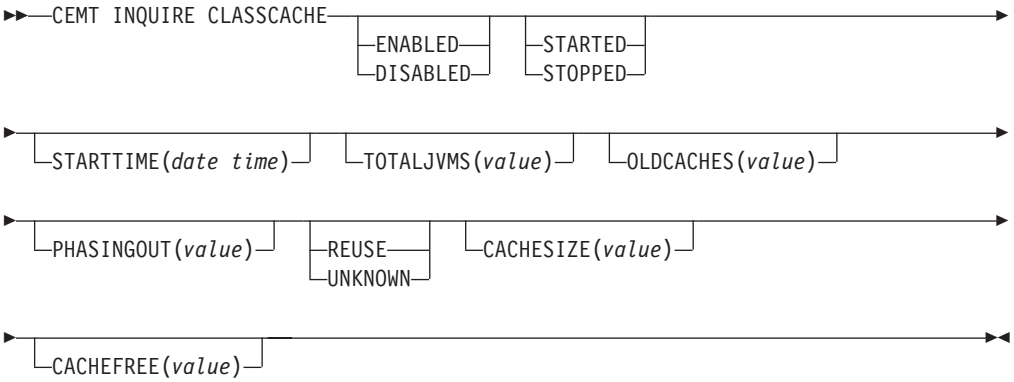
Input

Press the Clear key to clear the screen, and type `CEMT INQUIRE CLASSCACHE`. A screen is displayed that describes the active shared class cache and indicates any old shared class caches.

You can change the status of autostart by typing over the value for AUTOSTARTST. You can also change the displayed values by using CEMT commands:

- Change the status of autostart by using the **CEMT SET CLASSCACHE** command, or by entering `Enabled` or `Disabled` when you are performing the phase out, purge, or force purge operation with the **CEMT PERFORM CLASSCACHE** command.
- Change the size of the shared class cache, by using the `Cachesize` option when you are performing the start or reload operation with the **CEMT PERFORM CLASSCACHE** command. The new size that you specify is used for the new shared class cache.

INQUIRE CLASSCACHE



Displayed fields

AUTOSTARTST
Displays the status of autostart for the shared class cache. The values are as follows:

ENABLED

When autostart is enabled, if the shared class cache has been stopped or has not yet been started on this CICS run, the shared class cache is started as soon as CICS receives a request to run a Java application in a JVM with a profile that requires the use of the shared class cache.

DISABLED

When autostart is disabled, if the shared class cache has been stopped or has not yet been started on this CICS run, an explicit **PERFORM CLASSCACHE** command is required to start it. If the status of the shared class cache is STOPPED and autostart is disabled, and CICS receives a request to run a Java application in a JVM with a profile that requires the use of the shared class cache, the application cannot run.

CACHEFREE(*value***)**

Displays the amount of free space in the shared class cache, in bytes. CACHEFREE displays zero until the first JVM has run in the shared class cache.

CACHESIZE(*value***)**

Displays the size of the shared class cache, in bytes, kilobytes (K), megabytes (M), or gigabytes (G).

- If the status of the shared class cache is STOPPED, this value is the default size that is used when the shared class cache is started.
- If the status of the shared class cache is STARTING or STARTED, this value is the size of the current shared class cache.
- If the status of the shared class cache is RELOADING, this value is the size of the new shared class cache that is being loaded.

OLDCACHES(*value***)**

Displays the number of old shared class caches that are still present in the region because they are waiting for pooled JVMs that are using them to be phased out. If the status of the current shared class cache is STOPPED, and JVMs are still using it, that shared class cache is included in the number of old shared class caches.

PHASINGOUT(*value***)**

Displays the number of pooled JVMs that are using an old shared class cache and are being phased out. If the status of the current shared class cache is STOPPED, any JVMs that are still using it are included in the number of JVMs being phased out.

REUSEST

Returns a CVDA value indicating the reuse status of the pooled JVMs that use the shared class cache.

REUSE

The JVMs that use the shared class cache are continuous. In this CICS release, JVMs associated with the shared class cache are always continuous (reusable) JVMs.

UNKNOWN

The shared class cache is not started. Although CICS cannot identify the reuse status in this situation, the status always becomes REUSE when the shared class cache is started.

STATUS

Displays the status of the current shared class cache. The values are as follows:

STARTED

The shared class cache is ready and it can be used by pooled JVMs.

This value in the CEMT display includes both the status STARTED and the transient status RELOADING, which occurs when a new shared class cache is being loaded to replace the existing shared class cache. While the shared class cache is reloading, pooled JVMs that are using the shared class cache, both those that were already allocated to tasks and those that were allocated to tasks after the command was issued, continue to use the existing shared class cache until the new shared class cache is ready.

STOPPED
The shared class cache has either not been initialized on this CICS run, or it has been stopped by a **PERFORM CLASSCACHE** command or a **SET JVMPOOL** command. If autostart is disabled, requests to run a Java application in a pooled JVM with a profile that requires the use of the shared class cache fail. If autostart is enabled, a new shared class cache is initialized as soon as CICS receives a request to run a Java application in a JVM with a profile that requires the use of the shared class cache. This value in the CEMT display includes both the status STOPPED and the transient status STARTING, which occurs when the shared class cache is being initialized. While the shared class cache is starting, new pooled JVMs that require the use of the shared class cache wait until the startup process is complete and the shared class cache is ready. If initialization of the shared class cache is unsuccessful, any waiting requests for pooled JVMs that require the use of the shared class cache fail.

STARTTIME(*date time***)**
Displays the date and time on which the current shared class cache was started. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

TOTALJVMS(*value***)**
Displays the number of pooled JVMs in the CICS region that are using a shared class cache. This number includes both the JVMs that are using the current shared class cache and any JVMs that are using an old shared class cache and are being phased out.

CEMT INQUIRE CONNECTION

Retrieve information about an MRO or ISC over SNA connection to a remote system.

In the CICS Explorer, the ISC/MRO connections operations view provides a functional equivalent to this command.

Function

Description

CEMT INQUIRE CONNECTION returns information about a connection from your local CICS region to another CICS region or another system. The connection is either MRO or ISC over SNA. To inquire on connections that use TCP/IP, use the **CEMT INQUIRE IPCONN** command.

You can reset the options either by typing the SET command or by overtyping at the appropriate place on the INQUIRE screen.

See the *CICS Resource Definition Guide* for further information about connections.

The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key to clear the screen. There are two ways of starting this transaction:

- Type `CEMT INQUIRE CONNECTION`. You get a display that lists the current status.
- Type `CEMT INQUIRE CONNECTION` followed by as many of the other attributes as are necessary to limit the range of information that you require. So, for example, if you enter `cemt i c ins acq`, the resulting display will show you the details of only those connections that are in service and acquired.

You can then tab to the highlighted or blank fields and overtype them with the required values.

(*value***)**
specifies one or more names (1–4 characters) defined for an interregion communication (IRC) or an intersystem communication (ISC) connection.

ALL
is the default.

Sample screen

```
IN CONN
STATUS:  RESULTS - OVERTYPE TO MODIFY
  Con(G1GA)                  Ins     Irc Exci
  Con(ISC1) Net(AUHZCIC1)    Ins Acq Vta Appc     Xok Unk
                                        Nqn(GBIBMIYA.AUHZCIC1)
  Con(MRO2) Net(CICSHAA1)    Ins Acq Irc              Unk
  Con(MRO4) Net(CICSHAA2)    Ins Acq Irc              Unk
  Con(MRO5) Net(CICSHAA3)    Ins Acq Irc              Unk
  Con(MRO6) Net(CICSHAB1)    Ins Acq Xcf
  Con(VTA1) Net(IYCQTC03)    Ins Acq Vta Appc
                                        Nqn(GBIBMIYA.IYCQTC03)
```

Figure 49. CEMT INQUIRE CONNECTION screen

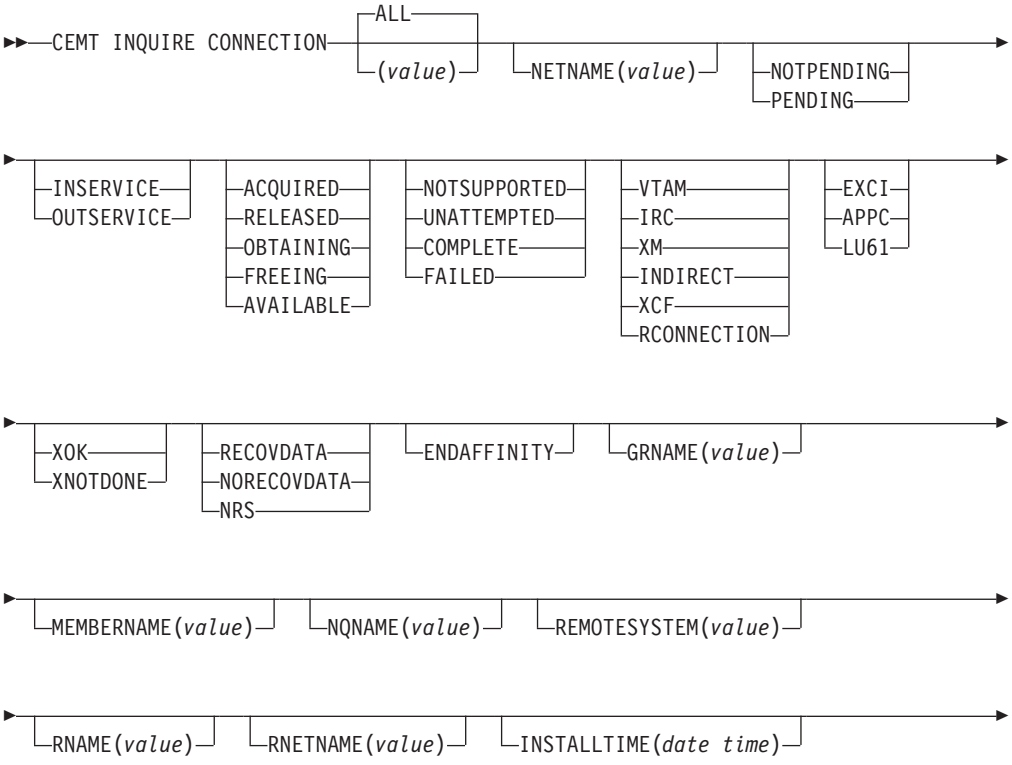
Note: There are blank fields on the screen where a value does not apply or is 'negative'; that is, the value begins with 'No'. To modify these fields, locate them by tabbing (they appear in the same sequence as in the expanded format), and overtype with input valid for that field. You may find it more convenient to use the expanded format when setting one of these values, (negating is easier because you are overtyping a displayed field with its negative causing the result to become non-displayed).

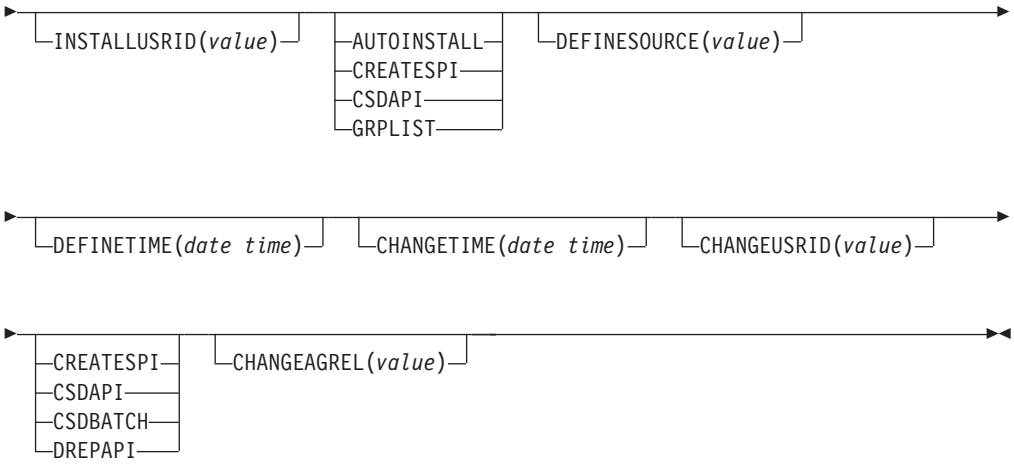
If you place the cursor against a specific entry in the list and press ENTER, CICS displays an expanded format as shown in Figure 50.

```
IN CONN
RESULT - OVERTYPE TO MODIFY
  Connection(MRO5)
  Netname(CICSHAA3)
  Pendstatus( Notpending )
  Servstatus( Inservice )
  Connstatus( Acquired )
  Accessmethod(Irc)
  Purgetype(            )
  Protocol()
  Xlnstatus()
  Recovstatus( Norecovdata )
  Uowaction(             )
  Cqp( Notsupported )
  Grname()
  Membername()
  Affinity(              )
  Nqname()
  Remotesystem()
  Rname()
  Rnetname()
```

Figure 50. The expanded display of an individual entry

CEMT INQUIRE CONNECTION





Displayed fields

ACCESSMETHOD
displays the access method in use for this connection. The values are:

VTAM
VTAM is now z/OS Communications Server. The connection is used for intersystem communication (ISC).

IRC The connection is used for multiregion operation (MRO), and has been defined to use the interregion communication (IRC) program for communication. If the CONNECTION is ACQUIRED, the MRO partner is running on the same MVS image. If the CONNECTION is RELEASED, the MRO partner may not be on the same MVS image; if it is not, the XCF access method will be used when the connection becomes ACQUIRED.

XM The connection is used for multiregion operation (MRO) and has been defined to use MVS cross-memory (XM) services for communication. If the CONNECTION is ACQUIRED, the MRO partner is running on the same MVS image. If the CONNECTION is RELEASED, the MRO partner may not be on the same MVS image; if it is not, the XCF access method will be used when the connection becomes ACQUIRED.

INDIRECT
Communication between the local CICS system and the system defined by this connection is through the system named in the INDSYS operand of the connection definition.

XCF The connection is used for multiregion operation (MRO), and communication uses the cross-system coupling facility (XCF) of MVS. XCF is used for MRO links between CICS regions on different MVS images within an z/OS sysplex. It is selected dynamically by CICS for such links when the access method is defined as IRC or XM in the connection definition.

RCONNECTION
The connection is remote—that is, it is owned by another CICS system.

AFFINITY (*input only field***)**
specifies, where CICS is a member of a z/OS Communications Server generic resource group, that z/OS Communications Server is to end an affinity owned by CICS. This option is valid only for APPC and LU61 connections. The connection must be out of service and in NORECOVDATA state. The value is:

ENDAFFINITY
z/OS Communications Server is to end an affinity owned by CICS.

CHANGEAGENT(*value***)**
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

AUTOINSTALL
The resource was autoinstalled.

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

CONNECTION(*value***)**
identifies this panel as relating to system connections, and displays the 4-character name by which the connection is known.

CONNSTATUS
displays the state of the connection between CICS and a remote system. The remote system can be an APPC or an MRO partner identified by (*value*) on the CEMT INQUIRE CONNECTION command. The ACQUIRED and RELEASED values are common to both APPC and MRO; the others are unique to APPC. The values are:

ACQUIRED (APPC and MRO)
The CONNECTION is acquired. The criteria for ACQUIRED for z/OS Communications Server links are:
- The partner LU has been contacted.
- Initial CNOS exchange has been done.

The criteria for ACQUIRED for MRO links are:
- Both sides of the link are in service.
- Both sides of the link are successfully logged on to the CICS interregion communication program (DFHIRP).
- A connection request by each side has been successful for at least one session, and therefore each side can send and receive data.

RELEASED (APPC and MRO)
The connection is released. Although the connection might be in service, it is not usable.

In the case of an MRO link, the released status can be caused by any one of a number of conditions. For example, it could be because the CICS region on the other side has not yet initialized, or not yet signed on to the CICS interregion communication program (DFHIRP); or it could be because CICS interregion communication may have been closed on the other side, or the connection on the other side may have been set out of service.

OBTAINING (APPC only)
The connection is being acquired. The connection remains in the OBTAINING state until all the criteria for ACQUIRED have been met.

FREEING (APPC only)
The connection is being released.

AVAILABLE (APPC only)
The connection is acquired but there are currently no bound sessions because they were unbound for limited resource reasons.

Connstatus can be reset with Acquired or Released.

CQP(*value***)**
displays the status of the connection quiesce protocol for the connection. The values are:

NOTSUPPORTED
The connection quiesce protocol is not supported for this connection.

UNATTEMPTED
CICS has not attempted to use the connection quiesce protocol (because the connection has not been released).

COMPLETE
The connection quiesce protocol completed succesfully when the connection was release. The status reverts to UNATTEMPTED when the connection is re-acquired.

FAILED
The connection quiesce protocol failed (for instance, becasue a session failed while the protocol was executing). The status reverts to UNATTEMPTED when the connection is re-acquired.

DEFINESOURCE(*value***)**
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

GRNAME*(value) (APPC only)*
displays (for an APPC connection to a generic resource when this system is also a generic resource) the 8-character generic resource name of the connected system. Otherwise this field is blank. CICS assumes that the partner is a generic resource if the two NETNAMEs sent with a BIND are different. Thus, information may be displayed for a partner that is *not* a generic resource but which uses XRF.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

AUTOINSTALL
The resource was autoinstalled.

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

MEMBERNAME(*value***)**
displays (for an APPC connection to a generic resource when this system is also a generic resource) the 8-character member name (applid) of the connected LU. Otherwise this field is blank. CICS assumes that the partner is a generic resource if the two NETNAMEs sent with a BIND are different. Thus, information may be displayed for a partner that is *not* a generic resource but which uses XRF.

NETNAME (*value***)**
displays the 8-character name by which the remote system is known to the network.

Note: If the netname is a z/OS Communications Server LU alias, it is different from the netname component of the NQNAME, which always contains the real netname.

NQNAME(*value***)**
displays the 17-character network-qualified name by which the remote system is known to the network.

PENDSTATUS *(APPC and MRO only)*
displays whether there are any pending units of work for this connection. The values are:
NOTPENDING
There has been no mismatch of lognames with the partner.

Note: MRO connections to pre-displays systems do not use lognames. Therefore, for these connections, PENDSTATUS is always NOTPENDING.
PENDING

There is resynchronization work outstanding for the connection but the partner system has performed an initial start, preventing completion of the resynchronization process. (If the partner system is a CICS version

earlier than CICS Transaction Server, a cold start performed on the partner system has the same effect.) You can use the SET CONNECTION NOTPENDING command to unilaterally commit or back out the units of work associated with the connection, according to their associated transaction definitions. You can also investigate the units of work individually and force them to commit or back out, in which case you must also complete the recovery activity by using a SET CONNECTION NOTPENDING command to clear the PENDING condition.

If this is an APPC connection, no new syncpoint work (that is, work involving synclevel 2 protocols) can be transmitted across it until a SET CONNECTION NOTPENDING command has been issued. This restriction does not apply to MRO connections.

If you are not concerned by the loss of synchronization caused by the initial (or cold) start of the partner, you can cause the SET CONNECTION NOTPENDING command to be issued automatically by specifying XLNACTION(FORCE) on the CONNECTION definition.

Pendstatus can be reset with Notpending.

For further information about pending units of work, see the *CICS Intercommunication Guide*.

PROTOCOL
displays the protocol in use if this is a z/OS Communications Server or external CICS interface connection. The values are:

APPC
The connection uses the z/OS Communications Server LUTYPE6.2 protocol for intersystem communication, or is remote.

EXCI The connection uses the external CICS interface for communication between CICS and a non-CICS client program.

LU61 The connection uses the z/OS Communications Server LUTYPE6.1 protocol for CICS-to-CICS or CICS-to-IMS intersystem communication.

PURGETYPE (*input only field***)**
specifies how associated transactions are to be purged. The values are:

PURGE (*z/OS Communications Server only and IRC only***)**
Transactions running on the connected system are abnormally terminated. Transactions are terminated only if system and data integrity can be maintained. A transaction is not purged if its definition specifies SPURGE=NO.

FORCEPURGE (*z/OS Communications Server only***)**
All transactions running on sessions on the connected system are immediately terminated abnormally. This can lead to unpredictable results and should be used only in exceptional circumstances.

In some extreme cases (for example, if an error occurs during backout processing), CICS might terminate abnormally.

CANCEL
Automatic initiate descriptors (AIDs) queuing for the specified connection are to be canceled.

AIDs representing scheduled and allocated requests waiting in the local CICS system for the specified connection are canceled. However, TD AIDs with an associated triggered task already started will not be canceled. In addition, the following CICS system AIDs are not purged unless FCANCEL is specified.

Table 2. System AIDs requiring FCANCEL to remove them

Description	Tranid
Remote delete AIDs	
Remote scheduler AIDs	CRSR
LU6.2 service manager 1 AIDs	CLS1
LU6.2 service manager 3 AIDs	CLS3
Remote scheduler PURGE AIDs	CRSQ
Resource manager resync AIDs	CRSY
Autoinstalled terminal delete AIDs	CATD
Autoinstalled terminal restart AIDs	CATR

When a canceled SCHEDULE request is found to have a precursor in a remote CICS system, that is, the AID was originally scheduled in a remote system, the remote AID is canceled asynchronously.

Message DFHTF0100 is written to CSMT to indicate how many AIDs have been deleted for the connection and how many remain.

An 'AIDS CANCELED' message appears on the CEMT panel whenever AIDs are deleted using the CANCEL option of the CEMT SET CONNECTION command.

FCANCEL
All AIDs, including system AIDs, queuing for the specified connection are to be canceled. See Table 2 for a list of those system AIDS that require FCANCEL to remove them. This can lead to unpredictable results and should be used only in exceptional circumstances.

Note: FCANCEL does not remove transient data AIDs with an associated triggered task. These aids may be removed by purging the associated task.

An 'AIDS CANCELED' message appears on the CEMT panel whenever AIDs are deleted using the FCANCEL option of the CEMT SET CONNECTION command.

RECOVSTATUS (*APPC and MRO only***)**
displays whether there is resynchronization work outstanding for this connection. The connection may never have been connected, or may have been quiesced and all resynchronization work completed, or disrupted without quiesce—in which case resynchronization may be necessary. The values are:

NORECOVDATA
Neither side has recovery information outstanding.

NRS CICS does not have recovery outstanding for the connection, but the partner may have.

RECOVDATA
There are indoubt units of work associated with the connection, or there are outstanding resyncs awaiting FORGET on the connection. Resynchronization takes place when the connection next becomes active, or when the UOW is unshunted.

If there is recovery outstanding, on completion of exchange of lognames either resynchronization takes place or, in the case of a cold exchange, the PENDING condition is created.

Recovstatus can be reset with Norecovdata.

REMOTESYSTEM (*value*)
displays the 4-character name of a connection, if the subject of the inquiry is a remote or an indirect connection. In either case, the named connection can be either a connection entry that links towards the TOR, or an indirect connection that provides the netname of the TOR.

If the subject of the inquiry is an indirect connection, the value of REMOTESYSTEM is taken from the INDSYS option of the CONNECTION definition.

Otherwise this field is blank.

RNAME (*value*)
displays the 4-character name by which this connection is known in a remote system, if the subject of the inquiry is a remote connection.

RNETNAME (*value*)
displays the 8-character netname of the owning TOR, if the subject of this inquiry is a remote connection. If it is blank, but the connection is remote, the system named in the REMOTESYSTEM field has not been installed, and no value was specified for the REMOTESYSNET option when the connection was defined.

SERVSTATUS
displays whether the system can receive and send data. The values are:
INSERVICE
The system is in service; that is, it is available for use.
OUTSERVICE
The system is out of service; that is, it is not available for use.

Note: You can reset this value by overtyping it with a different value.

UOWACTION (*input-only field*)
specifies that the normal resynchronization process is to be partially overridden. The value specifies the type of decision to be taken for any units of work that become indoubt due to a failure of the connection; the decisions are recorded and any data inconsistencies are reported when the connection is next acquired. The values are:
BACKOUT
All UOWs shunted due to the failure of this connection are to be backed out.
COMMIT
All UOWs shunted due to the failure of this connection are to be committed.
FORCEUOW
All UOWs shunted due to the failure of this connection are to be forced to BACKOUT or COMMIT, as specified on the INDOUBT option of the TRANSACTION definition.
RESYNC
Any UOWs shunted due to the failure of this connection are to be retried (that is, exchange lognames resynchronization for this connection is to be attempted). This process should normally be started automatically when a connection is acquired or when a UOW is unshunted.

XLNSTATUS
displays the status of the exchange lognames (XLN) process. The values are:

XOK (*APPC only***)**
The exchange log names (XLN) process for the APPC connection has completed successfully.

XNOTDONE (APPC only)
The exchange log names (XLN) flow for the APPC connection has not completed successfully. The CSMT log can contain information relating to this state. Synchronization level 2 conversations are not allowed on the connection, but synchronization levels 0 and 1 are still allowed.

No status is displayed if exchange log names is not applicable. This can be because the link:

- Is released
- Is MRO, LU6.1, or single-session APPC
- Does not support synchronization level 2 conversations.

For information about the APPC exchange log names process, see the *CICS Intercommunication Guide*.

CEMT INQUIRE CORBASERVER

Retrieves information about CorbaServers.

Description

INQUIRE CORBASERVER returns information about a particular CorbaServer.

The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

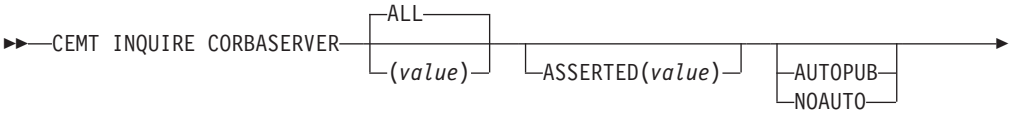
Input

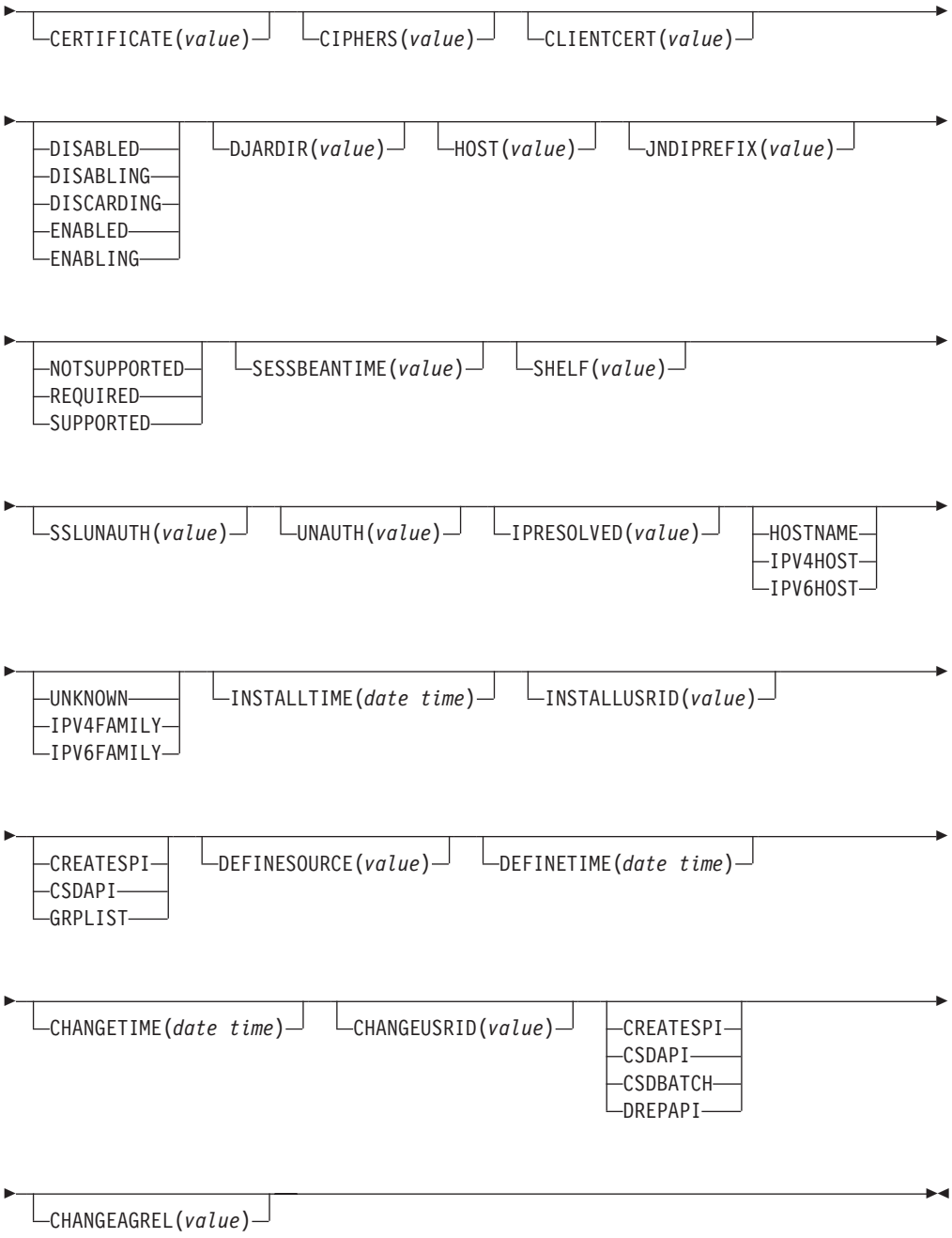
Press the Clear key to clear the screen, and type `CEMT INQUIRE CORBASERVER` (or suitable abbreviations for the keywords). The resulting display lists the names and status of the CorbaServer. You cannot change any of the displayed information.

(*value***)**
Specifies the 1- to 4-character name of the CorbaServer. The acceptable characters are A - Z a - z 0 - 9. Do not use names beginning with DFH, because these characters are reserved for use by CICS.

ALL
The default.

CEMT INQUIRE CORBASERVER





Displayed fields

ASSERTED(*value*)
Returns the 8-character name of a TCPIPSERVICE that defines the characteristics of the port that is used for inbound IIOP with asserted identity authentication.

AUTOPUBLISH(*value*)
Displays a value that indicates whether enterprise beans are to be automatically published to the JNDI namespace when the deployed JAR file that contains them is successfully installed in the CorbaServer. The possible values are as follows:

AUTOPUB
Enterprise beans are to be automatically published.

NOAUTO
Enterprise beans are not to be automatically published.

CERTIFICATE(*value***)**
Specifies the 1- to 56-character label of the certificate in the key ring that is to be used, as a client certificate, in the SSL handshake for outbound IIOP connections.

If this option is not specified, the default certificate for the key ring is used.

CIPHERS(*value***)**
Returns a 56-character area containing the list of cipher suites that is used to negotiate with clients during the SSL handshake. The list is set by the ENCRYPTION system initialization parameter, but you can edit the list to remove or change the order of cipher suites.

CLIENTCERT(*value***)**
Returns the 8-character name of a TCPIPSERVICE that defines the characteristics of the port that is used for inbound IIOP with SSL client certificate authentication.

CHANGEAGENT(*value***)**
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

DEFINESOURCE(*value***)**
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

DJARDIR(*value***)**
Displays the 255-character name of the deployed JAR file directory (also known as the *pickup directory*) on z/OS UNIX. The pickup directory is the place that you put deployed JAR files that you want to be installed into the CorbaServer by the CICS scanning mechanism.

ENABLESTATUS(*value***)**
Displays a value that indicates the current state of the CorbaServer. The possible values are as follows:

DISABLED
The CorbaServer is currently not processing any requests and cannot accept new requests. It might have failed to initialize properly or have been explicitly disabled.

DISABLING
The CorbaServer is quiescing before entering disabled state. It is not accepting new requests but is allowing current work to complete.

DISCARDING
A DISCARD request has been received for this CorbaServer. The CorbaServer is quiescing before being discarded. It is not accepting new requests but is allowing current work to complete.

ENABLED
The CorbaServer is available and is accepting requests.

ENABLING
The CorbaServer is being initialized. It is not yet ready to accept requests.

HOST(*value***)**
Displays the host name, IPv4, or IPv6 address of the logical EJB/CORBA server. All IPv4 addresses are displayed as native IPv4 dotted decimal addresses, for example, `1.2.3.4`, regardless of the type of address format used. For information on accepted IPv4 formats, see the *CICS Internet Guide*.

HOSTTYPE(*value***)**
Displays the address format of the HOST option. The HOSTTYPE option is available using an expanded query only. The possible values are as follows:

HOSTNAME
The HOST option contains a character host name.

IPV4HOST
The address is an IPv4 address.

IPV6HOST
The address is an IPv6 address.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

IPFAMILY(*value***)**
Displays the address format of the IPRESOLVED option. The IPFAMILY option is available using an expanded query only. Filtering interacts with the IPFAMILY option when you filter using wildcard characters. For example, if the value in IPFAMILY is IPV6FAMILY, you must use the colon character to retrieve an IP address. The possible values are as follows:

IPV4FAMILY
The address is an IPv4 address.

IPV6FAMILY
The address is an IPv6 address.

UNKNOWN
The IPRESOLVED option is not yet in use, or the address cannot be resolved.

IPRESOLVED(*value***)**
Displays the IPv4 or IPv6 address of the host. The IPRESOLVED option is available using an expanded query only.

JNDIPREFIX(*value***)**
Specifies a prefix of up to 255 characters to be used at run time when publishing enterprise beans to the Java Naming and Directory Interface (JNDI). The acceptable characters are A - Z a - z 0 - 9 . _ /.

The prefix must include any trailing delimiter, such as a forward slash, because CICS does not insert a delimiter between the prefix and a suffix.

If this option is not specified, no prefix is prepended when publishing beans to JNDI.

OUTPRIVACY(*value***)**
Displays the level of SSL encryption used for outbound requests from this CorbaServer. The level is determined by the CIPHERS attribute. Possible values are as follows:

NOTSUPPORTED
Encryption is not used. During the SSL handshake, CICS advertises only supported cipher suites that do not provide encryption.

REQUIRED
Encryption is used. During the SSL handshake, CICS advertises only supported cipher suites that provide encryption.

SUPPORTED
Encryption is used if both client and server support it. During the SSL handshake, CICS advertises all supported cipher suites.

SESSBEANTIME(*value***)**
Displays, in minutes, the period of inactivity after which a session bean might be discarded by CICS.

0 Session beans are not timed out.

vAlue Session beans might be discarded after the specified period of inactivity. The maximum value is 357839 (99 days, 23 hours, 59 minutes).

SHELF(*value***)**
Specifies the 1 - 255 character fully qualified name of a directory (a *shelf*, primarily for *deployed JAR files*) on z/OS UNIX. The acceptable characters are A - Z a - z 0 - 9 . _ /.

CICS regions into which the CORBASERVER definition is installed must have full permissions to the shelf directory. These permissions must include read, write, and the ability to create subdirectories.

A single shelf directory can be shared by multiple CICS regions and by multiple CORBASERVER definitions. Each CICS region uses a separate subdirectory to keep its files separate from those of other CICS regions. The subdirectories for CORBASERVER definitions are contained in the subdirectories of the CICS regions into which they are installed. When a CICS region performs a cold or initial start, it deletes its subdirectories from the shelf.

Do not modify the contents of a shelf that is referred to by an installed CORBASERVER definition. If you modify the contents of a shelf, the effects are unpredictable.

SSLUNAUTH(*value***)**
Returns the 8-character name of a TCPIPSERVICE that defines the characteristics of the port that is used for inbound IIOP with SSL and no authentication.

UNAUTH(*value***)**
Returns the 8-character name of a TCPIPSERVICE that defines the characteristics of the port that is used for inbound IIOP with no authentication.

CEMT INQUIRE DB2CONN

Retrieves information about the CICS DB2 connection.

In the CICS Explorer, the DB2 connections operations view provides a functional equivalent to this command.

Description

INQUIRE DB2CONN returns information about the attributes of the currently installed DB2CONN, which defines the connection to DB2.

Note that because only one DB2CONN can be installed at a time, the name of the DB2CONN is not required on input.

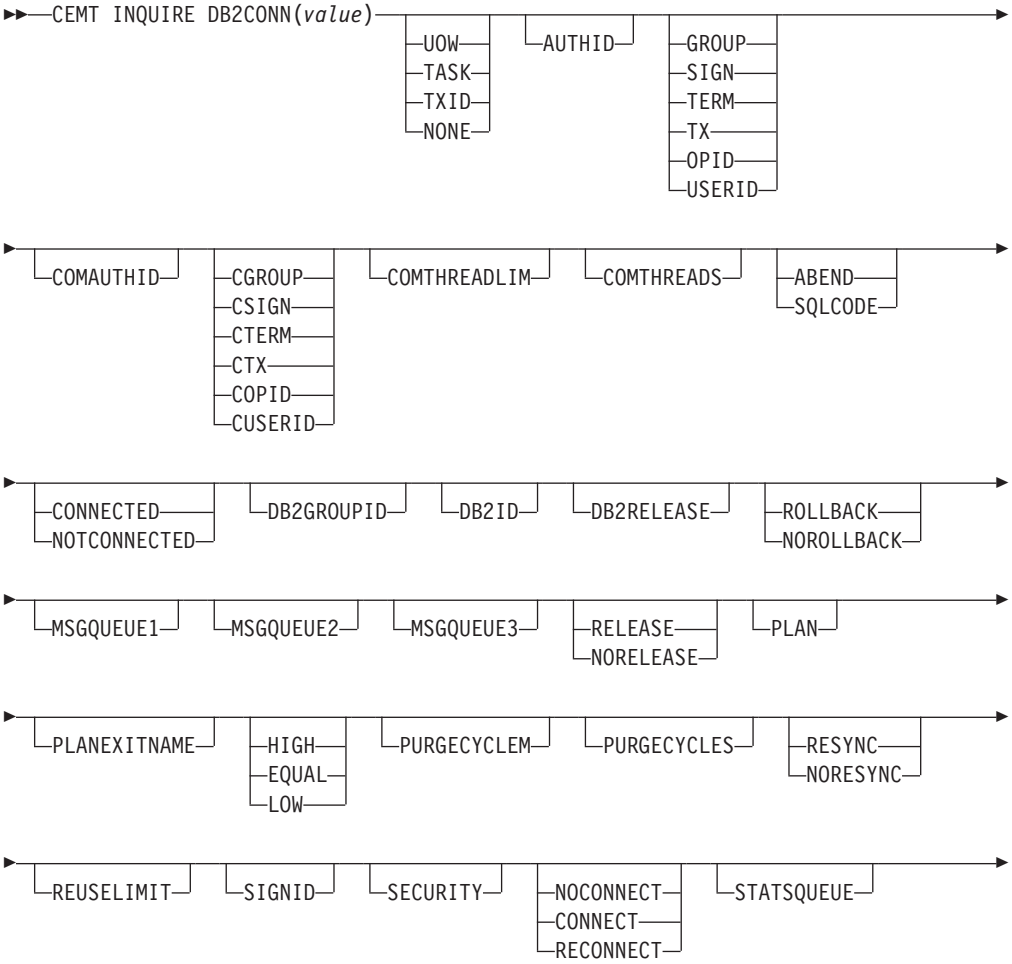
The resource signature

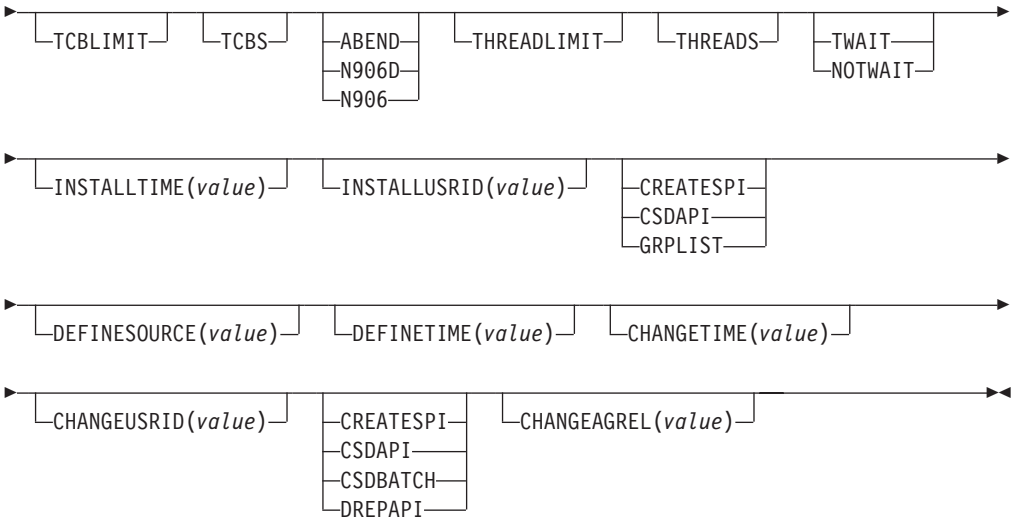
You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key to clear the screen, and type `CEMT INQUIRE DB2CONN`. The current status is displayed. You can then tab to the highlighted or blank fields and overtype them with the required values.

CEMT INQUIRE DB2CONN





Displayed fields

ACCOUNTREC
Returns the minimum amount of DB2 accounting required for transactions using pool threads. The specified minimum can be exceeded, as described in the following options. The values are as follows:

UOW The CICS DB2 attachment facility causes an accounting record to be produced for each UOW, assuming that the thread is released at the end of the UOW.

TASK
The CICS DB2 attachment facility causes a minimum of one accounting record to be produced for each CICS task.

A transaction containing multiple UOWs can use a different thread for each of its UOWs, assuming the thread is released at sync point. The result may be an accounting record produced for each UOW.

TXID The CICS DB2 attachment facility causes an accounting record to be produced when the transaction ID that is using the thread changes.

Because pool threads are typically used by a number of different transaction IDs, there is an increased chance that a transaction containing multiple UOWs will use a different thread for each UOW (assuming the thread is released at sync point. In this case, an accounting record can be produced for each UOW.

NONE
No accounting records are required for transactions using pool threads.

DB2 does, however, produce at least one accounting record for each thread when the thread is ended. Additionally, authorization changes cause accounting records to be produced.

AUTHID
Returns an ID to be used for security checking when using pool threads. If an AUTHID is returned, AUTHTYPE is not applicable.

AUTHTYPE
Returns the type of ID to be used for security checking when using pool threads. If an AUTHTYPE is returned, AUTHID is blank. The values are as follows:

GROUP
The 8-character user ID and the connected group name are used as the authorization ID. The following table shows how these two values are interpreted by DB2.

IDs passed to DB2	How DB2 interprets values
CICS sign-on user ID (USERID)	Represents the primary DB2 authorization ID.
RACF connected group name	If the RACF list of group options is not active, DB2 uses the connected group name supplied by the CICS attachment facility as the secondary DB2 authorization ID. If the RACF list of group options is active, DB2 ignores the connected group name supplied by the CICS attachment facility, but the value appears in the DB2 list of secondary DB2 authorization IDs.

If no RACF group ID is available for this user ID, an 8-character field of blanks is passed to DB2 as the group ID.

SIGN The SIGNID field of the DB2 connection is used as the resource authorization ID.

TERM
The terminal identification (four characters padded to eight) is used as an authorization ID. An authorization ID cannot be obtained in this manner if a terminal is not connected with the transaction.

If a transaction is started (using a CICS command) and has no terminal associated with it, AUTHTYPE(TERM) should not be used.

TX The transaction identification (four characters padded to eight) is used as the authorization ID.

OPID The user operator identification associated with the user ID, associated with the CICS transaction, is used as the authorization ID (three characters padded to eight).

USERID
The 8-character user ID associated with the CICS transaction is used as the authorization ID.

When the DB2 sample sign-on exit DSN3@SGN is used with AUTHTYPE(USERID), the exit sends the user ID to DB2 as the primary authorization ID and the RACF group ID to DB2 as the secondary ID. When the sample sign-on exit is used, there is no difference between AUTHTYPE(USERID) and AUTHTYPE(GROUP).

CHANGEAGENT(*value***)**
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

COMAUTHID
Returns an ID to be used for security checking when using command threads. If COMAUTHTYPE is returned, COMAUTHID is not applicable.

COMAUTHTYPE
Returns the type of ID to be used for security checking when using command threads. If COMAUTHTYPE is returned, COMAUTHID is blank. The values are as follows:

CGROUP
The 8-character user ID and the connected group name are used as the authorization ID. The following table shows how these two values are interpreted by DB2.

IDs passed to DB2	How DB2 interprets values
CICS sign-on user ID (USERID)	Represents the primary DB2 authorization ID.
RACF connected group name	If the RACF list of group options is not active, DB2 uses the connected group name supplied by the CICS attachment facility as the secondary DB2 authorization ID. If the RACF list of group options is active, DB2 ignores the connected group name supplied by the CICS attachment facility, but the value appears in the DB2 list of secondary DB2 authorization IDs.

If no RACF group ID is available for this user ID, an 8-character field of blanks is passed to DB2 as the group ID.

CSIGN
The SIGNID parameter of the DB2CONN is used as the resource authorization ID.

CTERM
The terminal identification (four characters padded to eight) is used as an authorization ID. An authorization ID cannot be obtained in this manner if a terminal is not connected with the transaction.

If a transaction is started (using a CICS command) and has no terminal associated with it, COMAUTHTYPE(CTERM) must not be used.

CTX The transaction identification (four characters padded to eight) is used as the authorization ID.

COPID
The operator identification associated with the user ID that is associated with the CICS transaction is used as the authorization ID (three characters padded to eight).

CUSERID
The 8-character user ID associated with the CICS transaction is used as the authorization ID.

When the DB2 sample sign-on exit DSN3@SGN is used with COMAUTHTYPE(CUSERID), the exit sends the user ID to DB2 as the primary authorization ID and the RACF group ID to DB2 as the secondary ID. When the sample sign-on exit is used, there is no difference between COMAUTHTYPE(CUSERID) and COMAUTHTYPE(CGROUP).

COMTHREADLIM
Returns the current maximum number of command threads that the CICS DB2 attachment allows to be active before requests overflow to the pool.

COMTHREADS
Returns the current number of active command threads.

CONNECTERROR
Returns a value when CICS is not connected to DB2 because the adapter is in standby mode. The value shows how this information is to be passed back to an application that has issued a SQL request. The values are as follows:
ABEND
The application is abended with abend AEY9.
SQLCODE
The application receives a -923 SQLCODE.

CONNECTST
Returns the status of the CICS DB2 connection. The values are as follows:
CONNECTED
CICS is connected to DB2.
NOTCONNECTED
CICS is not connected to DB2.

DB2GROUPID
If you are using group attach, this field returns the name (up to four characters) of a data sharing group, or subgroup of DB2 subsystems that you have specified. CICS tries to connect to any active member of this group. If the CICS DB2 attachment is connected, the name of the DB2 subsystem that was chosen from the group appears in the DB2ID field. If CICS is waiting to reconnect to a specific DB2 subsystem in the data sharing group for which it is holding outstanding units of work, the name of the specific DB2 subsystem appears in the DB2ID field, and CEMT displays the message `Waiting for DB2`. For this situation to arise, Resyncmember(Resync) must be specified.

DB2ID
If you are not using group attach, this field returns the name of the DB2 subsystem that the CICS DB2 attachment is connected to, or, if the CICS DB2 attachment is not connected, the name of the DB2 subsystem that you have specified for CICS to connect to. If you are using group attach and the CICS DB2 attachment is connected, this returns the name of the DB2 subsystem that the CICS DB2 attachment is connected to. If you are using group attach and the CICS DB2 attachment is not connected, this field is normally blank. However, if CICS is waiting to reconnect to a specific DB2 subsystem, because Resyncmember(Resync) is specified and the DB2 subsystem for which CICS is holding outstanding units of work is unavailable, this field gives the name of the DB2 subsystem, and CEMT displays the message `Waiting for DB2`.

DB2RELEASE
Returns a 4-character value indicating the version and release level of the DB2 subsystem to which CICS is connected. When CICS is not connected to DB2, blanks are returned.

DEFINESOURCE(*value***)**
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

DROLLBACK
Returns a value showing whether the CICS DB2 attachment is to initiate a sync point rollback if a transaction is adversely affected by a deadlock resolution. The values are as follows:

ROLLBACK
The attachment facility issues a sync point rollback before returning control to the application. An SQL return code of -911 is returned to the program.

NOROLLBACK
The attachment facility does not initiate a rollback for a transaction. An SQL return code of -913 is returned to the application.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

MSGQUEUE1
Returns the name of the first transient data destination to which unsolicited messages from the CICS DB2 attachment are sent.

MSGQUEUE2
Returns the name of the second transient data destination to which unsolicited messages from the CICS DB2 attachment are sent.

MSGQUEUE3
Returns the name of the third transient data destination to which unsolicited messages from the CICS DB2 attachment are sent.

NONTERMREL
Returns a value showing whether non-terminal transactions are to release threads for reuse at intermediate sync points:

RELEASE
Non-terminal transactions release threads for reuse at intermediate sync points.

NORELEASE
Non-terminal transactions do not release threads for reuse at intermediate sync points.

PLAN
Returns the name of the plan used for the pool. If a plan name is returned, PLANEXITNAME is blank,

PLANEXITNAME
Returns the name of the dynamic plan exit used for pool threads. If a PLANEXITNAME is returned, PLAN is blank,

PRIORITY
Returns the priority of the pool thread TCBs relative to the CICS main TCB (QR TCB). The thread TCBs are CICS open L8 TCBs. The values are as follows:

HIGH Thread TCBs have a higher priority than the CICS QR TCB.

EQUAL
Thread TCBs have equal priority with the CICS QR TCB.

LOW Thread TCBs have a lower priority than the CICS QR TCB.

PURGECYCLEM
Returns the number of minutes in the protected thread purge cycle time (the number of seconds is returned by PURGECYCLES). The range for PURGECYCLEM is 0 - 59.

A protected thread is not terminated immediately when it is released. It is terminated only after two completed purge cycles, if it has not been reused in the meantime. For example, if the protected thread purge cycle is set to 30 seconds, a protected thread is purged 30 - 60 seconds after that thread is released. An unprotected thread is terminated when it is released (at sync point or end of task) if there are no other transactions waiting for a thread on that DB2ENTRY.

PURGECYCLES
Returns the number of seconds in the protected thread purge cycle time (the number of minutes is returned by PURGECYCLEM). The range for PURGECYCLES is 0 - 59. If PURGECYCLEM is zero, the minimum value of PURGECYCLES is 5 seconds.

A protected thread is not terminated immediately when it is released. It is terminated only after two completed purge cycles, if it has not been reused in the meantime. For example, if the protected thread purge cycle is set to 30 seconds, a protected thread is purged 30 - 60 seconds after that thread is released. An unprotected thread is terminated when it is released (at sync point or end of task) if there are no other transactions waiting for a thread on that DB2ENTRY.

RESYNCMEMBER
This field applies only if you are using group attach (with a DB2 group ID specified), and returns the strategy that CICS adopts if outstanding units of

work are being held for the last DB2 data sharing group member to which CICS was connected. (Units of work that are shunted indoubt are not included in this process, because CICS itself cannot resolve those units of work at this time. Resynchronization for those UOWs occurs when CICS has resynchronized with its remote coordinator.) If you are not using group attach, a blank field is returned.

RESYNC
CICS connects to the same DB2 data sharing group member.

NORESYNC
CICS makes one attempt to connect to the same DB2 data sharing group member, and, if that attempt fails, CICS connects to any member of the DB2 data sharing group and issues a warning about the outstanding units of work.

REUSELIMIT
Returns a value in the range 0 - 10000 representing the maximum number of times a thread can be reused before it is terminated. The default is 1000. A value of 0 means that there is no limit on the number of times that a thread can be reused. Long-running CICS DB2 threads that are constantly being reused build up resources in DB2 that can cause storage problems.

The reuse limit applies to unprotected threads both in the pool and on a DB2ENTRY, and to protected DB2ENTRY threads.

SIGNID
Returns the authorization ID to be used by the CICS DB2 attachment when signing on to DB2 for pool and DB2 entry threads specifying AUTHTYPE(SIGNID) and command threads specifying COMAUTHTYPE(CSIGN).

SECURITY
Has no meaning for INQUIRE. It is supplied to allow overtyping with the REBUILD option, to initiate a CEMT SET DB2CONN. See SECURITY PURGECYCLES in **CEMT SET SECURITY**.

STANDBYMODE
Returns the action to be taken by the CICS DB2 attachment if DB2 is not active when an attempt is made to start the connection from CICS to DB2. The values are as follows:

NOCONNECT
The CICS DB2 attachment ends.

CONNECT
The CICS DB2 attachment goes into standby mode to wait for DB2.

RECONNECT
The CICS DB2 attachment goes into standby mode and waits for DB2. Having connected to DB2, if DB2 then fails, the CICS DB2 attachment reverts again to standby mode and later reconnects to DB2 when it starts again.

STATSQUEUE
Returns the transient data destination for CICS DB2 attachment statistics produced when the CICS DB2 attachment is shut down.

TCBLIMIT
Returns the maximum number of TCBs that can be used to process DB2 requests. CICS creates open TCBs, up to the limit specified in the system initialization parameter MAXOPENTCBS. The TCBLIMIT attribute of the

DB2CONN definition governs how many of the open TCBs can be used to access DB2; that is, how many of them can identify to DB2 and create a connection into DB2.

TCBS
Returns a number indicating the TCBs currently used by the CICS DB2 attachment facility. The TCBs used by the CICS DB2 attachment facility are allocated by CICS from the pool of L8 mode TCBs. A DB2 connection is not permanently assigned to the same L8 TCB, and, between CICS tasks, it can move from one L8 mode TCB to another. In this environment, the TCBs option returns the number of L8 mode TCBs that are using a DB2 connection at the time of the inquiry, and this value varies depending on workload.

THREADERROR
Returns the processing that is to occur following a create thread error. The values are as follows:

ABEND
For a second or subsequent SQL error, the transaction abends with abend code AD2S, AD2T, or AD2U, depending on the type of error that occurred. The transaction must be stopped and reinitialized before it is allowed to issue another SQL request.

N906D
A transaction dump is taken and the DSNCSQL RMI associated with the transaction is *not* to be disabled. The transaction receives a -906 SQLCODE error if another SQL is issued, unless the transaction issues SYNCPOINT ROLLBACK. SYNCPOINT without the ROLLBACK option results in an ASP3 or ASP7 abend. The transaction dump records an abend of AD2S, AD2T, or AD2U.

N906 The DSNCSQL RMI associated with the transaction is *not* to be disabled. The transaction receives a -906 SQLCODE error if another SQL request is issued, unless the transaction issues a SYNCPOINT ROLLBACK. SYNCPOINT without the ROLLBACK option results in an ASP3 or ASP7 abend.

THREADLIMIT
Returns the current maximum number of pool threads the CICS DB2 attachment allows to be active before requests are made to wait or are rejected. See Threadwait.

THREADS
Returns the current number of active pool threads.

THREADWAIT
Returns a value showing whether transactions wait for a pool thread or are abended if the number of active pool threads reaches the threadlimit number. The values are as follows:

TWAIT
If all threads are busy, a transaction waits until one becomes available.

NOTWAIT
If all threads are busy, a transaction is stopped with an abend code AD3T.

CEMT INQUIRE DB2ENTRY

Returns the attributes of the DB2ENTRY that defines resources to be used by a specific transaction or by a group of transactions when accessing DB2.

In the CICS Explorer, the DB2 entries operations view provides a functional equivalent to this command.

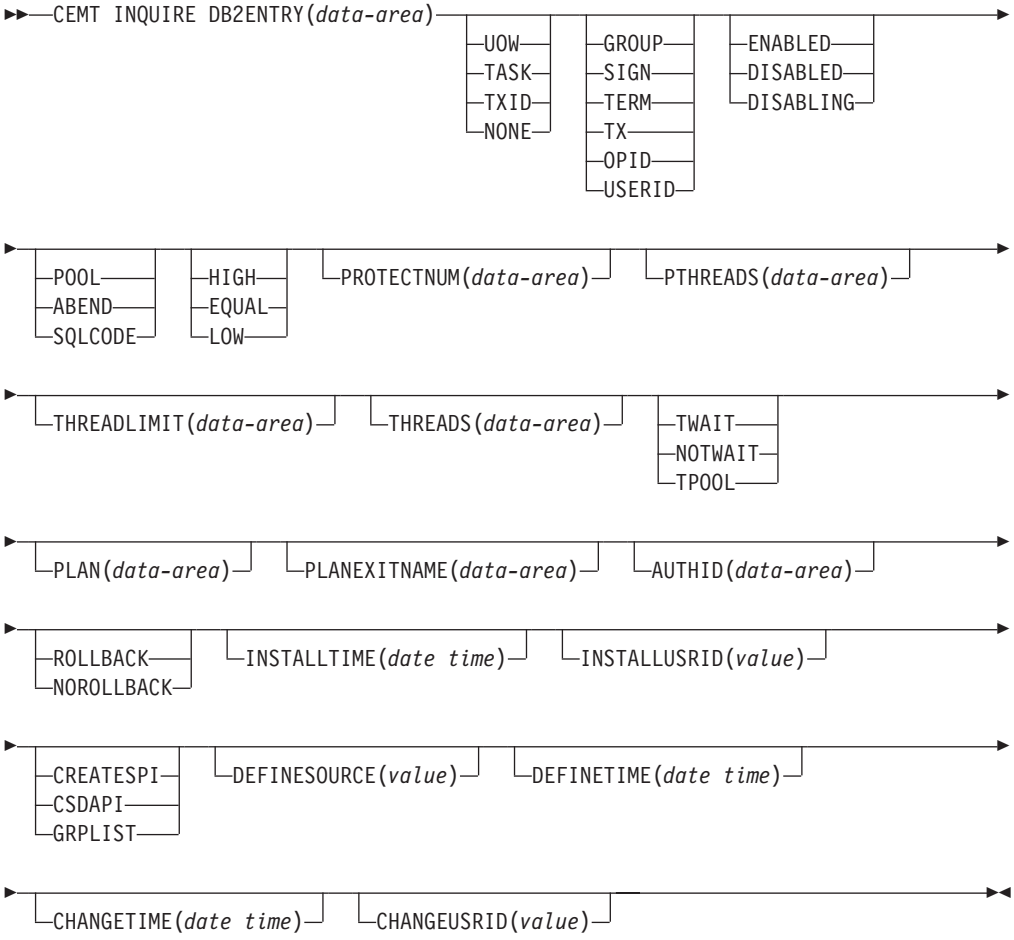
Description

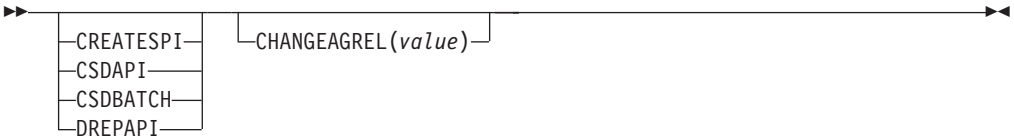
The entry is identified by the name with which it was defined in CEDA.

The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

CEMT INQUIRE DB2ENTRY





Displayed fields

ACCOUNTREC
Returns the minimum amount of DB2 accounting record required for transactions using this DB2ENTRY. The specified minimum might be exceeded, as described in the following options. The values are as follows:

UOW The CICS DB2 attachment facility causes an accounting record to be produced for each UOW, assuming that the thread is released at the end of the UOW.

TASK
The CICS DB2 attachment facility causes a minimum of one accounting record for each CICS task to be produced.

A transaction containing multiple UOWs may use a different thread for each of its UOWs (assuming the thread is released at sync point). The result may be an accounting record produced for each UOW.

TXID The CICS DB2 attachment facility causes an accounting record to be produced when the transid using the thread changes.

This option applies to DB2ENTRY inquiries that are used by more than one transaction ID. Because threads are typically released at sync point, a transaction containing multiple UOWs can use a different thread for each UOW. The result may be that an accounting record is produced for each UOW.

NONE
No accounting records are required for transactions using threads from this DB2ENTRY.

DB2 does, however, produce at least one accounting record for each thread when the thread is ended. Additionally, authorization changes cause accounting records to be produced.

AUTHID
Returns an ID to be used for security checking when using pool threads. If an AUTHID is returned, AUTHTYPE is not applicable.

AUTHTYPE
Returns the type of ID to be used for security checking for threads on this DB2ENTRY. If an AUTHTYPE is returned, AUTHID is blank. The values are as follows:

GROUP
The 8-character user ID and the connected group name are used as the authorization ID. The following table shows how these two values are interpreted by DB2.

IDs passed to DB2	How DB2 interprets values
CICS sign-on user ID (USERID)	Represents the primary DB2 authorization ID.

IDs passed to DB2	How DB2 interprets values
RACF connected group name	If the RACF list of group options is not active, then DB2 uses the connected group name supplied by the CICS attachment facility as the secondary DB2 authorization ID. If the RACF list of group options is active, DB2 ignores the connected group name supplied by the CICS attachment facility, but the value appears in the DB2 list of secondary DB2 authorization IDs.

If no RACF group ID is available for this user ID, an 8-character field of blanks is passed to DB2 as the group ID.

SIGN The **SIGNID** parameter of the DB2CONN is used as the resource authorization ID.

TERM
The terminal identification (four characters padded to eight) is used as an authorization ID. An authorization ID cannot be obtained in this manner if a terminal is not connected with the transaction.

If a transaction is started using a CICS command and has no terminal associated with it, do not use AUTHTYPE(TERM).

TX The transaction identification (four characters padded to eight) is used as the authorization ID.

OPID The operator identification associated with the user ID that is associated with the CICS transaction is used as the authorization ID (three characters padded to eight).

USERID
The 8-character user ID associated with the CICS transaction is used as the authorization ID.

When the DB2 sample sign-on exit DSN3@SGN is used with AUTHTYPE(USERID), the exit sends the user ID to DB2 as the primary authorization ID and the RACF group ID to DB2 as the secondary ID. When the sample sign-on exit is used, there is no difference between AUTHTYPE(USERID) and AUTHTYPE(GROUP).

CHANGEAGENT(*value*)
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

CHANGEAGREL(*value*)
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time*)
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value*)
Displays the 8-character user ID that ran the change agent.

DEFINESOURCE(*value*)
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time*)
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

DISABLEDACT
Returns a value showing what CICS is to do with new transactions accessing DB2ENTRY when it has been disabled or is disabling. If DISABLEDACT is not specified, and DB2ENTRY is disabled, new requests are routed to the pool by default. The values are as follows:

POOL
The CICS DB2 attachment facility routes the request to the pool. Message DFHDB2072 is sent to the transient data destination specified by MSGQUEUEn on the DB2CONN for each transaction routed to the pool.

ABEND
The CICS DB2 attachment facility stops the transaction.The abend code is AD26.

SQLCODE
An SQLCODE is returned to the application indicating that the DB2ENTRY is disabled.

DROLLBACK
Returns a value showing whether the CICS DB2 attachment should initiate a sync point rollback if a transaction is adversely affected by a deadlock resolution. The values are as follows:

ROLLBACK
The attachment facility issues a sync point rollback before returning control to the application. An SQL return code of -911 is returned to the program.

NOROLLBACK
The attachment facility does not initiate a rollback for this transaction. An SQL return code of -913 is returned to the application.

ENABLESTATUS
Indicates whether the DB2ENTRY can be accessed by applications. The values are as follows:

ENABLED
The DB2ENTRY can be accessed by applications. DB2ENTRY is installed in an ENABLED state.

DISABLED
The DB2ENTRY cannot be accessed by applications.

DISABLING
The DB2ENTRY is in the process of being disabled. New transactions

cannot access the DB2ENTRY. Existing transactions using the DB2ENTRY are allowed to complete unless the DB2ENTRY is being disabled with the FORCE option.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

PLAN
Returns the name of the plan to be used for this DB2ENTRY. If PLAN is returned, PLANEXITNAME is blank.

PLANEXITNAME
Returns the name of the dynamic plan exit (if any) to be used for this DB2ENTRY. If PLANEXITname is returned, PLAN is blank.

PRIORITY
Returns the priority of the pool thread TCBs relative to the CICS main TCB (QR TCB). The thread TCBs are CICS open L8 TCBs.
HIGH Thread TCBs have a higher priority than the CICS QR TCB.
EQUAL
Thread TCBs have equal priority with the CICS QR TCB.
LOW Thread TCBs have a lower priority than the CICS QR TCB.

PROTECTNUM
Returns the maximum number of protected threads allowed for this DB2ENTRY.

PTHREADS
Returns the current number of protected threads for this DB2ENTRY. A protected thread is an inactive thread available for reuse by a new transaction. If no transaction has reused the thread by the time it has been processed by two purge cycles, the thread is ended.

THREADLIMIT
Returns the current maximum number of threads for this DB2ENTRY that the CICS DB2 attachment allows active before requests are made to wait, overflow to the pool, or are rejected. See the Threadwait field.

THREADS
Returns the current number of threads active for this DB2ENTRY.

THREADWAIT
Indicates whether transactions wait for a DB2ENTRY thread to be abended, or overflow to the pool if the number of active DB2ENTRY threads reach the Threadlimit number. The values are as follows:

TWAIT
If all threads are busy, a transaction waits until one becomes available.

NOTWAIT
If any threads are busy, a transaction is stopped with an abend code AD2P.

TPOOL
If all threads are busy, a transaction is diverted to use a pool thread. If the pool is also busy, and NOTWAIT has been specified for the THREADWAIT parameter on the DB2CONN, the transaction is ended with abend code AD3T.

CEMT INQUIRE DB2TRAN

Returns attributes of a particular DB2TRAN that associates a transaction or group of transactions with a DB2ENTRY.

In the CICS Explorer, the DB2 transactions operations view provides a functional equivalent to this command.

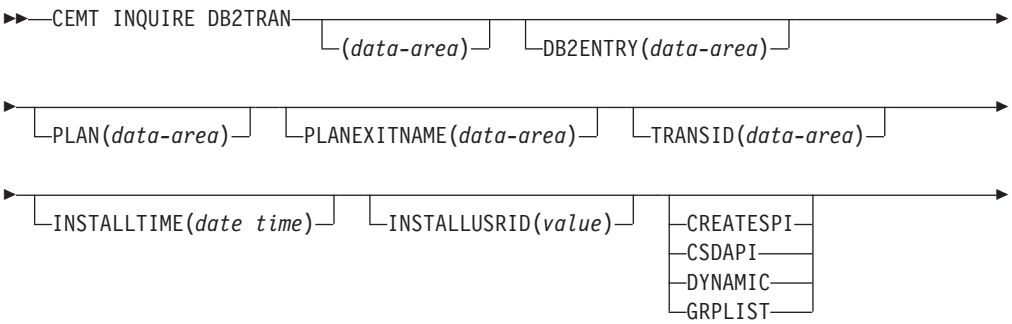
Description

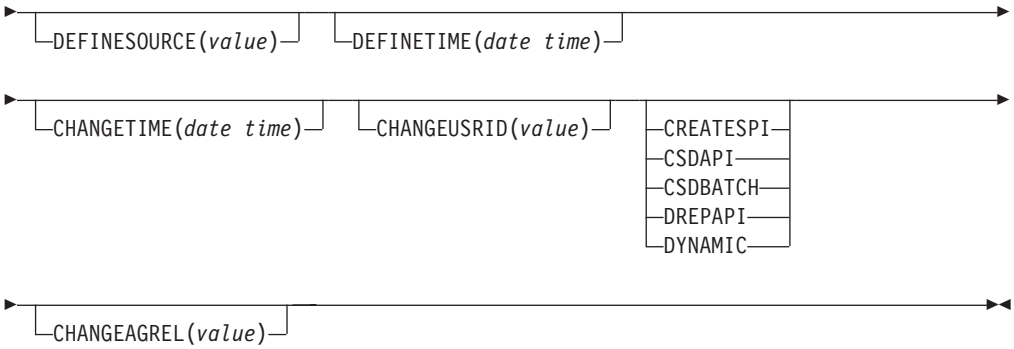
If a TRANSID is specified on a DB2ENTRY when the DB2ENTRY is installed, CICS installs a DB2TRAN named DFHtttt, where tttt is the TRANSID.

The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

CEMT INQUIRE DB2TRAN





Displayed fields

DB2ENTRY
Returns the name of the DB2ENTRY to which this DB2TRAN refers. This is the DB2ENTRY with which this additional transaction should be associated.

CHANGEAGENT(*value*)
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

DYNAMIC
The resource was defined as a result of the installation of a DB2ENTRY with TRANSID specified.

CHANGEAGREL(*value*)
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time*)
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value*)
Displays the 8-character user ID that ran the change agent.

DEFINESOURCE(*value*)
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

DYNAMIC
The resource was installed as a result of the installation of a DB2ENTRY with TRANSID specified.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

PLAN
Returns the name of the plan retrieved from the associated DB2ENTRY if it exists. If there is no DB2ENTRY, or the DB2ENTRY is disabled with DISABLEDACT(POOL), the pool plan name is returned if it exists. If PLAN is returned, PLANEXITNAME is blank.

PLANEXITNAME
Returns the name of the dynamic plan exit to be used, if any, from the associated DB2ENTRY if it exists. If there is no associated DB2ENTRY, or the DB2ENTRY is disabled with DISABLEDACT(POOL), the pool plan exit name is returned if it exists. If PLANEXITNAME is returned, PLAN is blank.

TRANSID
Specifies the transaction ID to be associated with the entry. The transaction ID can include wildcard characters. See the *CICS Resource Definition Guide*.

CEMT INQUIRE DELETSHIPPED

Display system settings that control the CICS timeout delete mechanism.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

The CICS timeout delete mechanism is invoked at user-specified intervals to remove any shipped terminal definitions that have not been used for longer than a user-specified time.

The **INQUIRE DELETSHIPPED** command displays the current settings of the parameters that control the mechanism.

Input

Press the Clear key and type CEMT INQUIRE DELETSHIPPED (or suitable abbreviations for the keywords). The resulting display is similar to the following example.

You can change attributes in the following ways:

- Overtype your changes on the INQUIRE screen after tabbing to the appropriate field.
- Use the CEMT SET DELETSHIPPED command.

Sample screen

```
IN DELE
STATUS:  RESULTS - OVERTYPE TO MODIFY
 Del Int( 120000 ) Idl( 020000 )
```

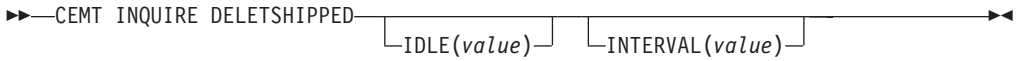
Figure 51. CEMT INQUIRE DELETSHIPPED screen

If you place the cursor against the result and press ENTER, CICS displays an expanded format as shown in Figure 52.

```
IN DELE
RESULT - OVERTYPE TO MODIFY
  Deletshipped
  Interval( 120000 )
  Idle( 020000 )
```

Figure 52. The expanded display

CEMT INQUIRE DELETSHIPPED



Displayed fields

DEL
Identifies this panel as relating to CICS timeout delete information.

IDLE(*value*)
Displays, in the form 0hhmmss+, the minimum time that an inactive shipped

terminal definition must remain installed in this region. When the CICS timeout delete mechanism is invoked, only those shipped definitions that have been inactive for longer than this time are deleted.

The time interval can be in the range 00 - 99 hours; 00 - 59 minutes; and 00 - 59 seconds.

INTERVAL(*value***)**
displays, in the form 0hhmmss+, the interval between invocations of the CICS timeout delete mechanism. The timeout delete mechanism removes any shipped terminal definitions that have not been used for longer than the time displayed by the IDLE option.

The time interval can be in the range 00 - 99 hours; 00 - 59 minutes; and 00 - 59 seconds.

CEMT INQUIRE DISPATCHER

Retrieve system information used by the CICS dispatcher.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Syntax

Press the Clear key and type `CEMT INQUIRE DISPATCHER`. You will get a display screen.

To change attributes, you can:

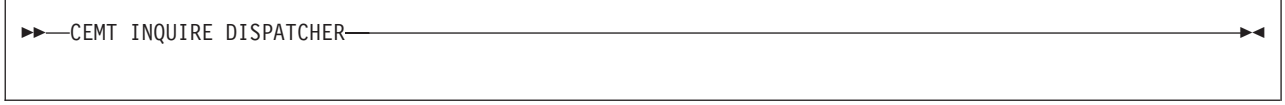
- Overtype your changes on the INQUIRE screen after tabbing to the appropriate field.
- Use the **CEMT SET DISPATCHER** command.

Sample screen

```
 inq dispatcher
  STATUS:  RESULTS - OVERTYPE TO MODIFY
Actjvmtcbs(000)
Actopentcbs(000)
Actssltcbs(000)
Actthrdtcbs(000)
Actxptcbs(000)
Aging( 32768 )
Maxjvmtcbs( 005 )
Maxopentcbs( 005 )
Maxssltcbs(008)
Maxthrdtcbs(012)
Maxxptcbs(005)
Mrobatch( 001 )
Runaway( 0005000 )
Scandelay( 0500 )
Subtasks(000)
Time( 0001000 )
                                    SYSID=HT61 APPLID=CICSHT61
   RESPONSE: NORMAL                TIME:  12.39.18  DATE: 07.30.01
 PF 1 HELP       3 END       5 VAR       7 SBH 8 SFH 9 MSG 10 SB 11 SF
```

Figure 53. CEMT INQUIRE DISPATCHER screen

CEMT INQUIRE DISPATCHER



Options

ACTJVMTCBS(*value***)**
Displays the total number of J8 and J9 mode open TCBs that are currently active; that is, allocated to a user task.

ACTOPENTCBS(*value***)**
Displays the number of L8 and L9 mode open TCBs that are currently active; that is, allocated to a user task.

ACTSSLTCBS(*value***)**
Displays the number of S8 mode open TCBS that are currently active; that is, allocated to a user task.

ACTTHRDTCBS(*value***)**
Displays the number of T8 mode open TCBs that are currently active; that is, allocated to a user task.

ACTXPTCBS(*value***)**
Displays the number of X8 and X9 mode open TCBs that are currently active; that is, allocated to a user task.

AGING(*value***)**
Displays the time factor for CICS to use in the priority aging algorithm for incrementing the priority of a task. The value is expressed as "milliseconds per unit of priority", and is in the range 0–65535.

The priority aging factor is defined initially by the **PRTYAGE** system initialization parameter, in which the value represents the number of milliseconds that must elapse before the priority of a waiting task can be adjusted upward by 1. See the *CICS System Definition Guide* for information about the **PRTYAGE** parameter.

Note: You can reset this value by overtyping it with a different value.

MAXJVMTCBS(*value***)**
Displays the maximum number of J8 and J9 mode open TCBs that CICS is allowed to attach. The value can be in the range 1–999.

Note: You can reset this value by overtyping it with a different value.

MAXOPENTCBS(*value***)**
Displays the maximum number of L8 and L9 mode open TCBs that can exist concurrently in the CICS region. The value can be in the range 1–2000.

Note: You can reset this value by overtyping it with a different value.

MAXSSLTCBS(*value***)**
Displays the maximum number of S8 mode open TCBs that can exist concurrently in the CICS region. The value can be in the range 1–1024.

Note: You can reset this value by overtyping it with a different value.

MAXTHRDTCBS(*value*)
Displays the maximum number of T8 mode open TCBs that can exist concurrently in the CICS region for all enabled JVMSERVER resources. The value can be in the range 1 - 1024.

MAXXPTCBS(*value*)
Displays the maximum number of X8 and X9 mode open TCBs that can exist concurrently in the CICS region. The value can be in the range 1–999.

Note: You can reset this value by overtyping it with a different value.

MROBATCH(*value*)
Displays the number of MRO requests from connected regions that are to be batched before this region is posted. It is a value in the range 1–255.

Note: You can reset this value by overtyping it with a different value.

RUNAWAY(*value*)
Displays the interval, in milliseconds, for which a task can have control before it is assumed to be looping.

Note: You can reset this value by overtyping it with a different value.

If you are resetting this value, you can specify 0, or a value in the range 500–2700000. The value you specify is rounded down to a multiple of 500.

CICS purges a task if the task has not given up control after this interval (that is, CICS assumes that the task is looping). If you specify zero, runaway task control is inoperative (that is, tasks are not purged if they appear to be looping). The value is independent of, and can be less than, the value in the TIME operand.

CICS runaway-task detection is based upon task time (that is, the interval is reset each time a task receives control of the processor, and is then decremented while the task is in control). You do not, therefore, need to allow for the effect of other jobs when setting the runaway value.

SCANDELAY(*value*)
Displays the maximum number of milliseconds between the receipt of a z/OS Communications Server request, and the time when CICS begins to process the request. The number is in the range 0–5000, and is less than or equal to the TIME value.

Note:

1. You can reset this value by overtyping it with a different value.
2. In the summary report of the dispatcher statistics, SCANDELAY is referred to as ICVTSD.

SUBTASKS(*value*)
Displays the value set by the SUBTSKS system initialization parameter, which can be either 0 or 1.

TIME(*value*)
Displays the interval, in milliseconds, for which CICS releases control to the operating system if no transactions are ready to resume processing. This interval is known as the ‘region exit interval'.

Note:

1. You can reset this value by overtyping it with a different value. It can have a value in the range 100–3600000, and must be greater than or equal to the SCANDELAY value.
2. In the summary report of the dispatcher statistics, TIME is referred to as ICV.

CEMT INQUIRE DJAR

Retrieves information about a DJAR definition.

Description

The **INQUIRE DJAR** command returns information about any currently installed DJAR definition.

The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key to clear the screen, and type `CEMT INQUIRE DJAR` (or suitable abbreviations for the keywords). The resulting display lists the names and status of any deployed JAR archive files. You cannot change any of the displayed information.

Sample screen

```
inq djar
STATUS:  RESULTS
 Djar(BJAR                             ) Corba(CORB) Inser
    Dates(20010221) Times(11:29:41) z/OS UNIXfi(/u/ivanh/ivanejbd.jar     )
 Djar(IJAR                             ) Corba(CORA) Inser
    Dates(20010222) Times(14:33:52) z/OS UNIXfi(/u/ivanh/ivanejbd.jar     )


                                          SYSID=HA61 APPLID=CICSHA61
  RESPONSE: NORMAL                       TIME:  15.54.22  DATE: 09.03.98
PF 1 HELP       3 END       5 VAR       7 SBH 8 SFH 9 MSG 10 SB 11 SF
```

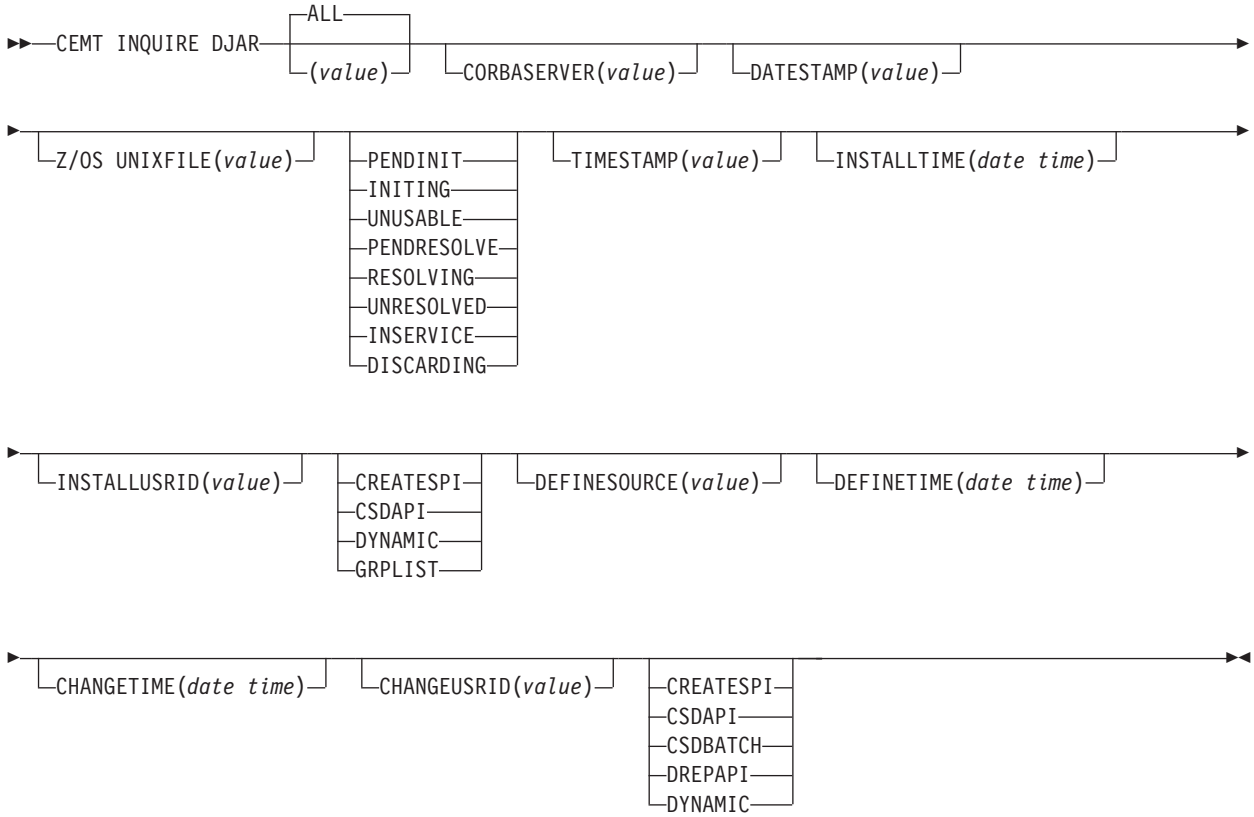
Figure 54. CEMT INQUIRE DJAR screen

If you place the cursor against a specific entry in the list and press ENTER, CICS displays an expanded format.

```
inq djar
RESULT
  Djar(BJAR)
  Corbaserver(CORB)
  State(Inservice)
  Datestamp(20010923)
  Timestamp(17:39:48)
  z/OS UNIXfile(/u/ivanh/ivanejbd.jar)
```

Figure 55. The expanded display of an individual entry

CEMT INQUIRE DJAR



(*value***)**
A 1- to 32-character deployed JAR file name.

ALL
The default.

Displayed fields

CHANGEAGENT(*value***)**
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

DYNAMIC
The resource was defined by the CICS scanning mechanism.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

CORBASERVER(*value***)**
Displays a 4-character CorbaServer name.

DATESTAMP(*value***)**
Displays the date, in yyyymmdd format, of the deployed JAR file on z/OS UNIX. This is a read-only value that CICS updates when the DJAR resource is installed or updated. You can use the DATESTAMP and TIMESTAMP values to determine whether CICS has refreshed itself after you update a JAR in the deployed JAR file (pickup) directory.

For dynamically-installed DJARs (those installed by the CICS scanning mechanism), the value of DATESTAMP is the date of the z/OS UNIX file pointed to by the DJAR definition, at the time the DJAR definition was last installed or updated.

For statically-installed DJARs (those installed from a CSD or by CREATE DJAR), the value of DATESTAMP is the date of the z/OS UNIX file pointed to by the DJAR definition, at the time the DJAR was installed.

If the INQUIRE DJAR command is executed before the DJAR resource has fully initialized, this field displays blanks.

DEFINESOURCE(*value***)**
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

DYNAMIC
The resource was installed by the CICS scanning mechanism.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

Z/OS UNIXFILE(*value***)**
Displays a 255-character, fully-qualified z/OS UNIX file name.

STATE(*value***)**
The values are as follows:

PENDINIT
The DJar has just been created and is awaiting Stage 1 Resolution.

INITING
The DJar is undergoing Stage 1 Resolution processing (or has got stuck in this phase). See EJIO, which involves copying the DJar to the shelf.

UNUSABLE
The DJar has failed Stage 1 Resolution and is not available for use.

PENDRESOLVE
The DJar has undergone Stage 1 Resolution correctly (or was recovered on a warm start after resolving correctly in an earlier region run) and is awaiting Stage 2 Resolution processing (Bean loading).

RESOLVING
The DJar is undergoing Stage 2 Resolution Processing (or has got stuck in this state, see EJJO) and so its Beans are being loaded by CEJR.

UNRESOLVED
The DJar has failed Stage 2 Resolution, or some other processing, and is not available for use. Beans were not correctly loaded.

INSERVICE
The DJar has undergone both Stage 1 and Stage 2 Resolution and is available for use. Its Beans are all correctly loaded.

DISCARDING
The DJar is being deleted.

TIMESTAMP(*value*)

Displays the time, in hh:mm:ss format, of the deployed JAR file on z/OS UNIX. This is a read-only value that CICS updates when the DJAR resource is installed or updated. You can use the DATESTAMP and TIMESTAMP values to determine whether CICS has refreshed itself after you've made an update to a JAR in the deployed JAR file (pickup) directory.

For dynamically-installed DJARs (those installed by the CICS scanning mechanism), the value of TIMESTAMP is the time of the z/OS UNIX file pointed to by the DJAR definition, at the time the DJAR definition was last installed or updated.

For statically-installed DJARs (those installed from a CSD or by CREATE DJAR), the value of TIMESTAMP is the time of the z/OS UNIX file pointed to by the DJAR definition, at the time the DJAR was installed.

If the INQUIRE DJAR command is executed before the DJAR resource has fully initialized, this field displays blanks.

CEMT INQUIRE DOCTEMPLATE

Retrieves information about a DOCTEMPLATE resource definition for a CICS document template.

In the CICS Explorer, the Document templates operations view provides a functional equivalent to this command.

Description

INQUIRE DOCTEMPLATE returns information about any currently installed document templates.

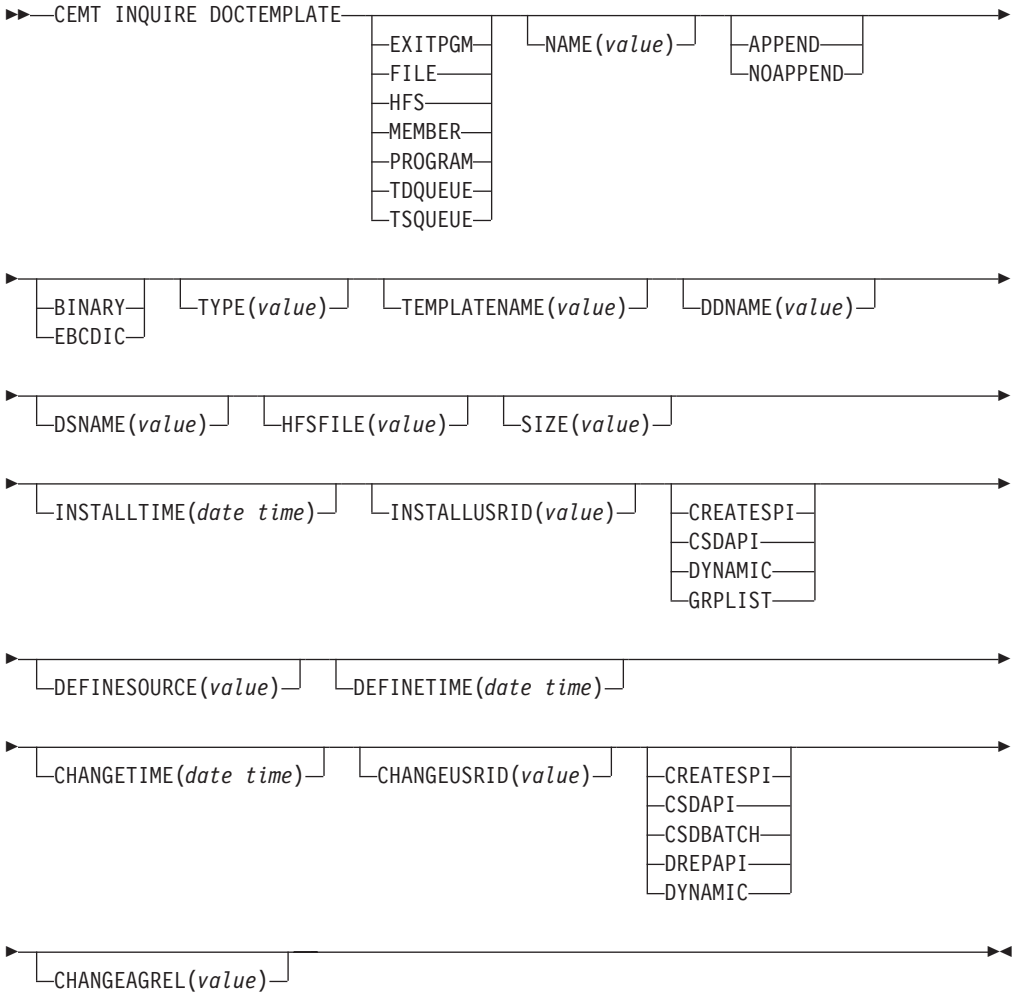
The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key to clear the screen, and type CEMT INQUIRE DOCTEMPLATE (or suitable abbreviations for the keywords). The resulting display lists the names and status of any document templates. You cannot change the displayed information.

CEMT INQUIRE DOCTEMPLATE



Displayed fields

APPENDCRLF
Returns whether CICS deletes trailing blanks from, and appends carriage-return line-feed to, each logical record of the template.

APPEND
Append carriage-return and line-feed.

NOAPPEND
Do not append carriage-return and line-feed.

CHANGEAGENT(*value*)
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

DYNAMIC
The resource was defined by the CICS system for a template being used through the CICS template manager, DFHWBTL.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

DDNAME
Returns the DD name of the PDS containing the template. The DD name applies only to a template of type PDS. If a member name is supplied without a value for DDNAME, the default value DFHHTML is used.

DEFINESOURCE(*value***)**
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

DOCTEMPLATE
Returns the name of this DOCTEMPLATE resource definition.

DSNAME
Returns the DS name of the PDS containing the template.

HFSFILE
Returns the fully qualified name of the z/OS UNIX file in which the template resides. The HFSFILE attribute applies only to a template of type HFS. It can be up to 255 characters in length.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

DYNAMIC
The resource was installed by the CICS system for a template being used through the CICS template manager, DFHWBTL.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

NAME
Returns the name of the location defined in TEMPLATETYPE.

SIZE
Returns the amount of storage, in bytes, used by the cached copy of the document template. A value of zero is returned if there is no cached copy of the template at the time of the inquiry.

TEMPLATENAME
Returns the extended template name by which the document template is known outside the resource definition function (the TEMPLATENAME attribute of the DOCTEMPLATE resource definition).

TEMPLATETYPE
Returns the type of resource with the name returned in NAME.

EXITPGM
An exit program.

FILE A CICS file name for a data set.

HFS A z/OS UNIX file.

MEMBER
A name of the member in the PDS described in DDNAME.

PROGRAM
A program.

TDQUEUE
A TD queue.

TSQUEUE
A TS queue.

TYPE
Returns the format of the template contents.

BINARY
When the template is loaded from the template library, no parsing of the template contents takes place.

EBCDIC
When the template is loaded from the template library, the contents are parsed as EBCDIC text.

CEMT INQUIRE DSAS

Retrieve information about the local CICS dynamic storage areas (DSAs).

In the CICS Explorer, the Global Dynamic Storage Areas operations view provides a functional equivalent to this command.

Description

CEMT INQUIRE DSAS returns the value of various DSA sizes and limits.

Input

Press the Clear key and type CEMT INQUIRE DSAS (or suitable abbreviations for the keywords).

You can change various attributes in the following ways:

- Overtype your changes on the INQUIRE screen after tabbing to the appropriate field (see Figure 56).
- Use the **CEMT SET DSAS** command.

Sample screen

```
IN DSAS
STATUS:  RESULTS - OVERTYPE TO MODIFY
   Sosabovebar(Notsos)
   Sosaboveline(Notsos)
   Sosbelowline(Notsos)

  Dsalimit( 04194304)
   Cdsasize( 00524288 )
   Rdsasize( 00262144 )
   Sdsasize( 00262144 )
   Udsasize( 00000000 )

  Edsalimit( 0052428800)
   Ecdsasize( 0002097152 )
   Erdsasize( 0003145728 )
   Esdsasize( 0000000000 )
   Eudsasize( 0001048576 )

  Memlimit(Nolimit)
```

Figure 56. CEMT INQUIRE DSAS screen

CEMT INQUIRE DSAS

```
►►—CEMT INQUIRE DSAS———————————————————————————————————————►◄
```

Displayed fields

CDSASIZE(*value*)
Displays the size, in bytes, of the CICS dynamic storage area (CDSA). CICS calculates and manages the size of this storage area automatically, within the overall limits specified for all the DSAs that reside below 16 MB (below the line).

DSALIMIT(*value*)
Displays the maximum amount of storage, as a total number of bytes, within which CICS can dynamically allocate storage for the four individual DSAs that reside below 16 MB (below the line). For more information, see DSALIM system initialization parameter in the System Definition Guide.

Note: You can reset this value by overtyping it with a different value.

If DSALIMIT specifies a value lower than the current limit, CICS might not implement the new limit immediately, but attempts to do so over time as dynamic storage is freed in the individual DSAs.

ECDSASIZE(*value***)**
Displays the size, in bytes, of the extended CICS dynamic storage area (ECDSA) above 16 MB but below 2 GB (above the line). CICS calculates and manages the size of this storage area automatically, within the overall limits specified for all the DSAs that reside above the line.

EDSALIMIT(*value***)**
Displays the maximum amount of storage, as a total number of bytes, within which CICS can dynamically allocate storage for the individual DSAs that reside above 16 MB but below 2 GB (above the line). For more information, see EDSALIM system initialization parameter in the System Definition Guide.

You can reset this value by overtyping it with a different value.

If EDSALIMIT specifies a value lower than the current limit, CICS might not implement the new limit immediately, but attempts to do so over time as dynamic storage is freed in the individual DSAs.

ERDSASIZE(*value***)**
Displays the size, in bytes, of the extended read-only dynamic storage area (ERDSA) above 16 MB but below 2 GB (above the line). CICS calculates and manages the size of this storage area automatically, within the overall limits specified for all the DSAs that reside above the line.

ESDSASIZE(*value***)**
Displays the current size of the extended shared dynamic storage area (ESDSA) above 16 MB but below 2 GB (above the line). CICS calculates and manages the size of this storage area automatically, within the overall limits specified for all the DSAs that reside above the line.

EUDSASIZE(*value***)**
Displays the size, in bytes, of the extended user dynamic storage area (EUDSA) above 16 MB but below 2 GB (above the line). CICS calculates and manages the size of this storage area automatically, within the overall limits specified for all the DSAs that reside above the line.

MEMLIMIT(*value***)**
Displays the limit of storage above the bar, for use by the CICS region. A value of NOLIMIT indicates that no limit is imposed on the amount of storage that the region can attempt to use.

RDSASIZE(*value***)**
Displays the current size of the read-only dynamic storage area (RDSA). CICS calculates and manages the size of this storage area automatically, within the overall limits specified for all the DSAs that reside below 16 MB (below the line).

SDSASIZE(*value***)**
Displays the current size of the shared dynamic storage area (SDSA). CICS calculates and manages the size of this storage area automatically, within the overall limits specified for all the DSAs that reside below 16 MB (below the line).

SOSABOVEBAR(*value***)**
Displays whether CICS is short on storage in the dynamic storage areas above the bar.

NOTSOS
CICS is not short on storage in any of the dynamic storage areas above the bar.

SOS CICS is short on storage in at least one of the dynamic storage areas above the bar.

SOSABOVELINE(*value***)**
Displays whether CICS is short on storage in the dynamic storage areas above 16 MB but below 2 GB (above the line).

NOTSOS
CICS is not short on storage in any of the dynamic storage areas above 16 MB but below 2 GB.

SOS CICS is short on storage in at least one of the dynamic storage areas above 16 MB but below 2 GB.

SOSBELOWLINE(*value***)**
Displays whether CICS is short on storage in the dynamic storage areas below 16 MB (below the line).

NOTSOS
CICS is not short on storage in any of the dynamic storage areas below 16 MB.

SOS CICS is short on storage in at least one of the dynamic storage areas below 16 MB.

UDSASIZE(*value***)**
Displays the size, in bytes, of the user dynamic storage area (UDSA) below 16 MB (below the line).

CEMT INQUIRE DSNAME

Retrieve information about an external data set.

Description

The **INQUIRE DSNAME** command returns information about the status of a data set managed by CICS file control. The data set name is either the name specified on the DSN parameter of the DD statement of the data set, or the DSNAME attribute of the FILE resource definition of the data set. You can use this command to monitor failing VSAM data sets. Many of the attributes cannot be determined until a file that references the data set has been opened. If this has not happened, blank values are returned for most attributes.

Note: CEMT has to retrieve QUIESCESTATE information from the ICF catalog. This can slow down processing of this command. QUIESCESTATE is not supplied by default but is retrieved and displayed if requested.

Input

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT INQUIRE DSNAME` (or suitable abbreviations for the keywords). The resulting display the current status except for the QUIESCESTATE value. To retrieve the QUIESCESTATE value, place the cursor at the beginning of the displayed item (in front of the Dsn field) and press the Enter key.
- Type `CEMT INQUIRE DSNAME` (or suitable abbreviations for the keywords), followed by the attributes that are necessary to limit the range of information that you

require. For example, if you enter `cemt i dsn vs not`, the resulting display shows the details of only those external data sets that are VSAM data sets and are not recoverable.

You can tab to the highlighted or blank fields and overtype them with the required values.

ALL
The default.

(*value***)**
Specifies one or more names (1 - 44 characters) of an external data set.

Sample screen

```
IN DSN
STATUS:  RESULTS - OVERTYPE TO MODIFY
 Dsn(CICSTS42.CICS.CICSH###.DFHCSD                    ) Vsa
    Fil(0001) Val Bas     Sta         Ava
 Dsn(CICSTS42.CICS.CICSHT61.FILEA                     ) Vsa
    Fil(0001) Val Bas     Sta         Ava
```

Figure 57. CEMT INQUIRE DSNAME screen

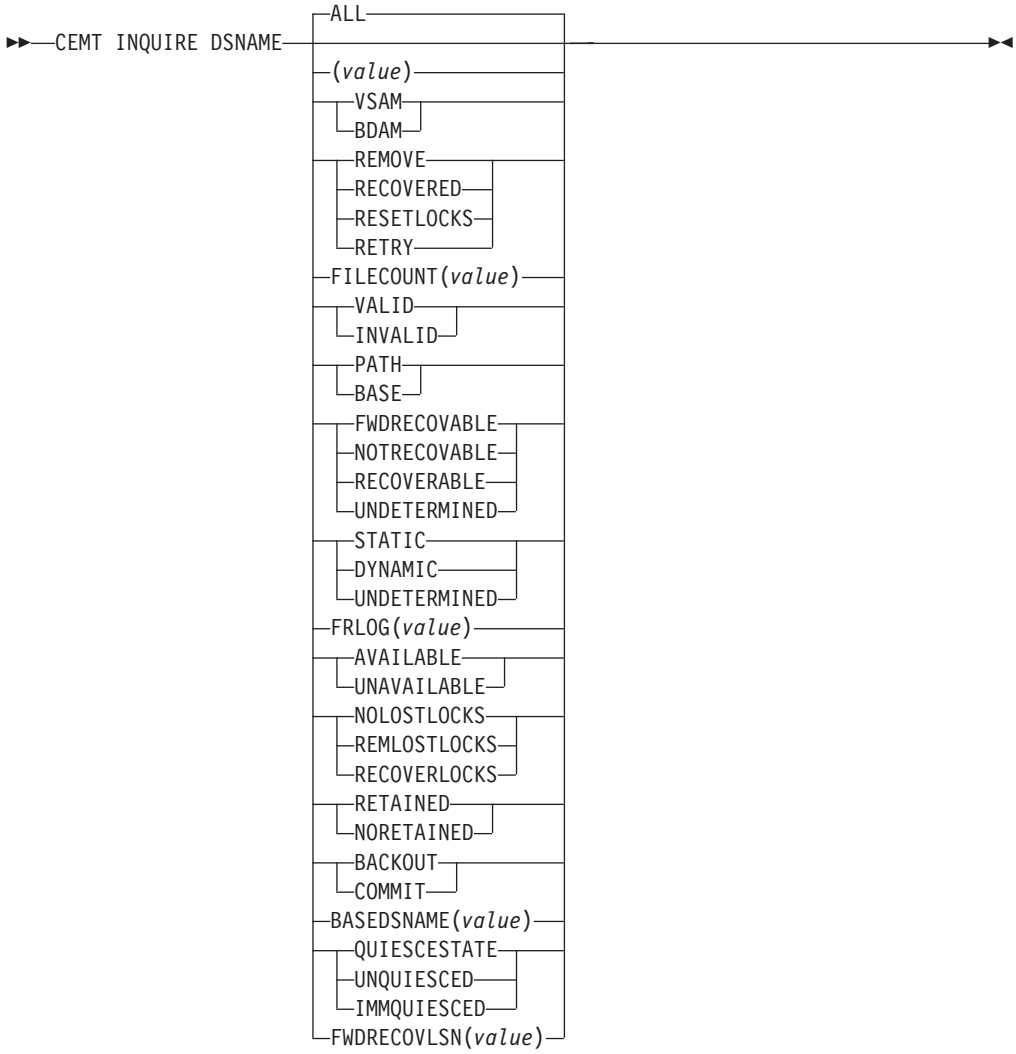
Note: When a value does not apply, or is negative (the value begins with No), the fields on the screen are blank. To modify these fields, locate them by tabbing (they appear in the same sequence as in the expanded format), and overtype with input valid for that field. You might find it more convenient to use the expanded format when setting one of these values.

Because of the large number of keywords supported for this command, only two lines of information for each result are displayed. The full set of keywords can be displayed by moving the cursor to the beginning of a displayed item (in front of the Dsn field) and pressing the Enter key. The resulting display shows all the keyword information for the displayed item, as follows:

```
IN DSN
RESULT - OVERTYPE TO MODIFY
  Dsname(CICSTS42.CICS.CICSH###.DFHCSD)
  Accessmethod(Vsam)
  Action(              )
  Filecount(0001)
  Validity(Valid)
  Object(Base)
  Recovstatus(Notrecovable)
  Backuptype(Static)
  Frlog()
  Availability( Available )
  Lostlocks()
  Retlocks(Noretained)
  Uowaction(              )
  Basedsname(CICSTS42.CICS.CICSH###.DFHCSD)
  Quiescestate( Unquiesced )
  Fwdrecovlsn()
```

Figure 58. The expanded display of an individual entry

CEMT INQUIRE DSNAME



Displayed fields

ACCESSMETHOD
Displays the access method of the data set. The values are as follows:

VSAM
Virtual storage access method (VSAM).

BDAM
Basic direct access method (BDAM).

ACTION (*input only field***)**
Specifies the action to take on this data set. The values are as follows:

REMOVE
Use this option when a data set is no longer required on the local system. It deletes the data set name block for the named data set. The REMOVE option can be used only when there are no file control resource definition entries that refer to the specified data set. If any file definitions refer to the data set name when you enter SET DSNAME REMOVE, you receive an error message indicating that the use count

of the data set name block is not zero. Use the following procedure to ensure that no file definitions refer to the data set name:

1. Close all files that refer to the data set. You can use the following command to find these files:

   ```
   CEMT INQUIRE FILE(*) DSNAME(dsname)
   ```

2. Either remove the data set name from each file entry (by over-typing the data set name with blanks on the CEMT display), or discard the file definition entirely.

If you have a base data set with one or more alternate indexes, you cannot remove the data set name block for the base until you have removed all references to the base data set and all references to paths.

RECOVERED

The data set has been restored from a backup version and forward recovery has run and completed successfully. CICS normally updates the backup-while-open (BWO) attributes for the base cluster data set BASEDSNAME, using MVS/DFP 3.2 Callable Services.

If the BWO attributes are not updated after the restore of a backup copy of a data set that was taken using BWO, CICS fails a subsequent file open because the data set is marked as down-level.

RESETLOCKS (*VSAM only***)**

Purges shunted unit of work (UOW) log records for backout-failed and commit-failed UOWs that hold locks on this data set, and releases the retained locks:

- Backout-failed UOWs are those that failed during backout processing.
- Commit-failed UOWs are those that have updated record-level sharing (RLS) data sets, and have failed to release locks during the second phase of 2-phase commit syncpoint processing.

Use this option only as a last resort, because you are accepting backout failure and some loss of data integrity as an alternative to retaining locks and delaying transactions.

For backout-failed and commit-failed UOWs that hold locks on the data set, all records relating to this data set are removed from the system log and all retained record locks held by this CICS for the data set are released. Diagnostic messages are written to the CSFL transient data queue for each backout-failed log record that is removed as a result of the RESETLOCKS operation.

The RESETLOCKS option applies only to shunted UOW log records that are classed as backout-failed and commit-failed. Therefore it cannot be issued for a BDAM data set. Backout-failed and commit-failed log records are not shunted for BDAM data sets.

You might use RESETLOCKS if backout-failed or commit-failed log records are holding up lost locks recovery for the data set, and there is no other way to resolve them.

Note:

1. This option does not apply to shunted *indoubt* UOWs. Try to resolve the shunted indoubt UOWs that hold locks on the data set in other ways before issuing RESETLOCKS; for example, by using COMMIT, BACKOUT, or FORCE (see the CEMT SET DSNAME command for information about these options).

2. RESETLOCKS can fail during the commit phase (for example, if an error occurs while CICS is trying to release the RLS locks), in which case the UOWs revert to being shunted as commit-failed UOWs.

RETRY

Specifies that shunted UOW log records, caused by failed backout and commit processing as a result of the failure of this data set, should be retried. This is similar in operation to the **EXEC CICS RESYNC** command, but applies only to backout-failed and commit-failed UOWs, and not to indoubt UOWs.

Use RETRY when the data set has shunted backout- or commit-failed UOWs associated with it, and you believe that some or all of the data set problems are either transient or have been resolved.

Messages, issued at the time of a data set failure that causes UOWs to be shunted, recommend the actions that are required to recover from the failure.

RETRY does not affect data integrity, and can be used safely at any time to enable some failed recovery work to complete.

AVAILABILITY

Displays information indicating whether the data set is currently flagged, in this CICS region, as available or unavailable for use. The availability indicator is a local flag that a CICS region maintains in a data set name block (DSNB) for each data set. If the file is a BDAM data set, no information is returned. The values are as follows:

AVAILABLE

The data set is available for use according to the CICS data set name block. CICS can issue either RLS or non-RLS open requests for this data set.

Note: Although a data set is available according to information held by CICS, an open request could still fail if the ICF catalog indicates otherwise. This can occur, for example, if data set recovery is pending or in progress.

UNAVAILABLE

The data set is marked by CICS as not available for use. The CICS region is unable to open the data set in either RLS or non-RLS mode.

AVAILABILITY can be reset by specifying AVAILABLE.

BACKUPTYPE

Displays the backup type of the data set. The values are as follows:

STATIC

The data set is accessed in non-RLS mode and is not eligible for BWO processing. All the files in this data set must be closed and the data set serialized by a backup manager such as DFSMS/MVS before a data set backup copy can be made.

DYNAMIC

The data set is accessed in RLS mode and is eligible for BWO processing. Data set backup is allowed while it is open for output, using a data set backup manager such as DFSMShsm or DFSMSdss. Hierarchical storage manager (DFSMShsm) and data set services (DFSMSdss), which perform data set backup and recovery operations, are components of IBM Data Facility Storage Management Subsystem (DFSMS/MVS) program product.

UNDETERMINED
The data set has not been opened by the CICS region in which the command is issued, or the data set is BDAM or a VSAM PATH. NOTAPPLIC is also returned if the data set has been opened in RLS mode. The VSAM catalog should be referred to get the BWO status.

BASEDSNAME (*value*) (VSAM ONLY)
When PATH is displayed as well, this is the name of the base data set with which this path is associated.

When BASE is displayed as well, this is the same as the name shown in the Dsn(*value*) field.

DSNAME (*value*)
Indicates that this panel relates to an external data set inquiry and displays the 44-character name of an external data set.

FILECOUNT (*value*)
Displays information about how many different files are currently associated with this data set. You can associate a file with a data set by a job control DD statement, or dynamically by a DSNAME parameter on the file definition. If a file has been associated by a job control DD statement, the file name cannot be changed by a CEDA command, CEMT, or an **EXEC CICS SET FILE DSNAME** command.

FILECOUNT does not show how many files are open against the data set; it only shows how many files are associated with the data set. If a file is associated with a data set and is then closed and associated with a different data set, the FILECOUNT of the first data set decreases by 1, and the FILECOUNT of the second data set increases by 1.

FRLOG (*value*)
Displays the numeric journal identifier of the journal being used to log the updates for forward recovery if this data set is forward recoverable (FWDRECOVABLE). The FRLOG value is undefined if the data set is not forward recoverable. You can define a data set as forward recoverable either in the ICF catalog or, if it is accessed in non-RLS mode, in the file resource definition. This option is meaningful only for data sets accessed in LSR or NSR mode, and for which the forward recovery attributes are obtained from the CICS file resource definition.

For data sets accessed in RLS mode (if DFSMS 1.3 is installed), or non-RLS mode data sets for which CICS obtains the recovery attributes from the ICF catalog, CICS displays a value of zero.

FWDRECOVLSN (*value*)
Displays the name of the log stream that is used to log the updates if this data set is defined as forward recoverable. CICS returns nulls if the data set is not forward recoverable.

The log stream name returned is either of the following:

- The log stream name specified directly in the ICF catalog
- For a non-RLS access mode data set that does not have forward recovery attributes in the ICF catalog, it is a log stream name identified by CICS through a journal name generated from the FWDRECOVLOG value.

LOSTLOCKS
Displays whether there are any lost locks for this data set. The possible values displayed are as follows:

NOLOSTLOCKS
The data set has no lost locks.

REMLOSTLOCKS
The data set has lost locks, therefore is unavailable, but no recovery is required on this CICS system. The lost locks belong to a another (remote) CICS region.

RECOVERLOCKS
The data set has lost locks and therefore is unavailable. This CICS system is performing lost-locks recovery.

OBJECT
Displays the type of the data set. The values are as follows:

PATH (VSAM only)
This data set is a path leading to another data set.

BASE The named data set is a base data set, rather than a path data set.

QUIESCESTATE
Displays the RLS quiesce state of the data set. The information is obtained from the ICF catalog entry for the data set. The values displayed are as follows:

QUIESCED
A data set has been quiesced. It means that RLS access method control blocks (ACBs) cannot open against the data set and no CICS region has an RLS ACB currently open against this data set. However, the data set can be opened in non-RLS mode.

UNQUIESCED
This is the normal value for a data set that is not quiescing or has not been quiesced. It indicates that RLS or non-RLS ACBs can be opened against the data set, the mode being established by the first open. After an ACB is opened in one mode, other ACBs can be opened only in the same mode.

If a data set is quiescing because a **SET DSNAME QUIESCE** command was issued in the same region as the **CEMT I DSN** command, the words "BEING QUIESCED" are displayed.

If a data set is quiescing, the words "BEING QUIESCED" are displayed for the CICS region in which the command was issued.

If quiesce does not apply (for example, for a BDAM data set), nothing is displayed.

Quiescestate can be reset by specifying Quiesced | Unquiesced or:

IMMQUIESCED
Causes all existing CICS open RLS ACBs throughout the sysplex to be closed and the data set to be marked as quiesced in the ICF catalog. Each CICS in the sysplex abends all in-flight UOWs that are accessing the data set before closing ACBs, causing in-flight UOWs to back out. Any OUWs that fail backout are shunted. No RLS ACBs can open against this data set, but non-RLS open requests are permitted.

RECOVSTATUS
Displays the recovery status of the data set. The values are as follows:

FWDRECOVABLE
All updates to the data set are logged for both backout and forward recovery.

NOTRECOVABLE
Updates to the data set are not logged.

This response may also be returned as the result of use of the XFCNREC global user exit. A program enabled at XFCNREC may indicate that file opens should proceed even if there is a mismatch in the backout recovery requirements for different files associated with same data set. In these circumstances, the data set is marked as NOTRECOVABLE to indicate that its data integrity can no longer be guaranteed.

The condition remains until cleared by a CEMT or **EXEC CICS SET DSNAME REMOVE** command, or by an initial or cold start of CICS (if the associated data set is not in backout-failed state). While the data set is in this state, backout logging is performed for a particular request based on the specification in the file definition. Therefore backout logging may occur for requests via one file and not via another.

RECOVERABLE
All updates to the data set are logged for backout.

UNDETERMINED
The recovery status of the data set is not known because files accessing the data set have not been opened for output.

RETLOCKS
Displays whether there are any retained record locks, as a result of deferred recovery work by this CICS region, for the specified data set.

RETAINED
This CICS region has deferred recovery work causing retained locks for the data set. One effect of this is that, if the data set was last opened in RLS mode, the locks are RLS locks and therefore the data set cannot be opened in non-RLS mode.

Another effect is that any FILE definitions that specify this data set cannot be changed to specify a different data set.

If the data set is a BDAM data set, or a VSAM data set accessed in non-RLS mode, the locks are CICS record locks, otherwise they are RLS record locks.

NORETAINED
This CICS region has no deferred recovery work for the base data set, and therefore no retained locks.

Note: NORETAINED can also mean that the CICS region is currently processing deferred recovery work. When the recovery work is successfully completed, the value remains as NORETAINED until more deferred work is created. If the recovery work is not successfully completed, the value reverts to RETAINED.

UOWACTION (*input only field***)**
Specifies the action to take for all shunted units of work that reference this data set. The values are as follows:

BACKOUT
Specifies that all shunted indoubt units of work that hold locks on the data set should be backed out.

COMMIT
Specifies that all shunted indoubt units of work that hold locks on the data set should be committed. Specifies that all shunted indoubt units of work that hold locks on the data set should be forced to BACKOUT or COMMIT, as specified by the ACTION attribute on the transaction resource definition.

VALIDITY
Displays the validity of the data set name. The values are as follows:
VALID
The data set name has been validated against the VSAM catalog by opening a file against that data set.
INVALID
The data set name has not been validated against the VSAM catalog by opening a file against that data set.

CEMT INQUIRE DUMPDS

Retrieve information about the CICS dump data sets.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

You can use the **INQUIRE DUMPDS** command to view which dump data set (A or B) is currently open.

Instead of using the SET command, you can overtype your changes on the INQUIRE screen after tabbing to the appropriate field. See "Overtyping a display" on page 117.

For information on using dumps, see the *CICS Problem Determination Guide*.

Input

Press the Clear key and type `CEMT INQUIRE DUMPDS` (or suitable abbreviations for the keywords).

The resulting display is similar to the following example. To change any of the values shown, tab to the highlighted or blank fields and overtype them as required.

Sample screen

```
IN DUMPDS
STATUS:  RESULTS - OVERTYPE TO MODIFY
 Dum Cur(A) Ope
```

Figure 59. CEMT INQUIRE DUMPDS screen

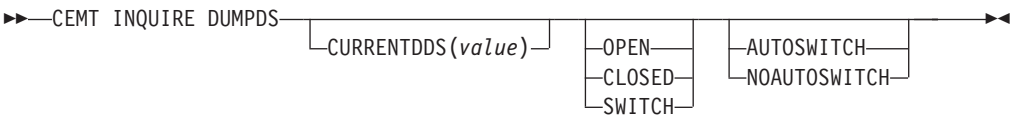
Note: When a value does not apply, or is negative (the value begins with No), the fields on the screen are blank. To modify these fields, locate them by tabbing (they appear in the same sequence as in the expanded format), and overtype with input valid for that field. You might find it more convenient to use the expanded format when setting one of these values.

If you place the cursor against the result and press ENTER, CICS displays an expanded format as shown in Figure 60 on page 204.

```
IN DUMPD
RESULT - OVERTYPE TO MODIFY
  Dumpds
  Currentdds(A)
  Openstatus( Open )
  Switchstatus( Noautoswitch )
```

Figure 60. The expanded display

CEMT INQUIRE DUMPDS



Displayed fields

CURRENTDDS (*value*)
Displays a one character indicator that shows which dump data set is active. This can be A or B

DUMPDS
Indicates that this panel relates to a dump data set inquiry.

OPENSTATUS
Display the status of the active CICS dump data set. The values are as follows:

OPEN
The active CICS dump data set is open.

CLOSED
The active CICS dump data set is closed.

SWITCH
The currently active dump data set is to become inactive, and the currently inactive data set is to become active.

Note: You can reset this value by overtyping it with a different value.

SWITCHSTATUS
Display whether there is an automatic switch to the inactive dump data set when the active dump data set is full. The values are as follows:

AUTOSWITCH
The switch occurs only when the active dump data set is full, but not subsequently.

NOAUTOSWITCH
When the active dump data set is full, there is no automatic switch to the inactive dump data set.

Note: Because this is a negative attribute, the field appears blank. You can, however, tab to this field and overtype it with a different value.

Note: You can reset this value by overtyping it with a different value.

CEMT INQUIRE ENQ

Retrieve information about enqueues held or waited on by a unit of work (UOW), or about UOWs holding or waiting on a specified enqueue.

In the CICS Explorer, the Unit of work enqueues operations view provides a functional equivalent to this command.

Description

The **INQUIRE ENQ** command returns information about the enqueues held by a UOW. Enqueues are used by CICS to lock recoverable resources such as file records or queues to the UOW that is updating them. User enqueues obtained by the EXEC CICS ENQ command are also returned.

You can use **INQUIRE ENQ** to indicate which records and queues would be affected if the UOW were forced.

The transaction also returns information about UOWs that are waiting on the enqueues. you can use this to diagnose enqueue deadlocks between tasks that want to update the same resources.

Enqueues are typically held in active state, which means that other tasks are allowed to wait for the enqueue. However, if the UOW that owns the enqueue suffers an indoubt failure, the enqueue is usually converted to the retained state until the indoubt failure can be resolved. **INQUIRE ENQ** also retrieves information about these enqueues and can be used to identify which records and queues would be affected if the UOW were forced.

Note:

1. Both UOW-lifetime and task-lifetime enqueues are returned by **INQUIRE ENQ**. (For an explanation of UOW- and task-lifetime enqueues, see the MAXLIFETIME option of the **EXEC CICS ENQ** command.).
2. On an indoubt failure, user enqueues are released, *unless* the **EXEC CICS ENQ** command specified MAXLIFETIME(TASK) and it is not the end-of-task syncpoint that suffers the failure.

Input

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT INQUIRE ENQ` (or suitable abbreviations for the keywords). The resulting display lists all enqueues for all the current UOWs.
- Type `CEMT INQUIRE ENQ` (or suitable abbreviations for the keywords), followed by the attributes that are necessary to limit the range of information that you require. For example, if you enter `cemt i enq d`, the resulting display shows the details only of enqueues on records in VSAM data sets open in non-RLS mode or CICS-maintained data tables.

Sample screen

```
INQ ENQ
STATUS:  RESULTS
 Enq(AC143001D4A72601) Tra(CECI) Tas(0000038) Act Exe Own
    Res(ACCOUNTS.DBASE1          ) Rle(015) Enq(00000000)
 Enq(AC143001D4A72601) Tra(CECI) Tas(0000038) Act Add Own
    Res(g...                     ) Rle(004) Enq(00000000)
```

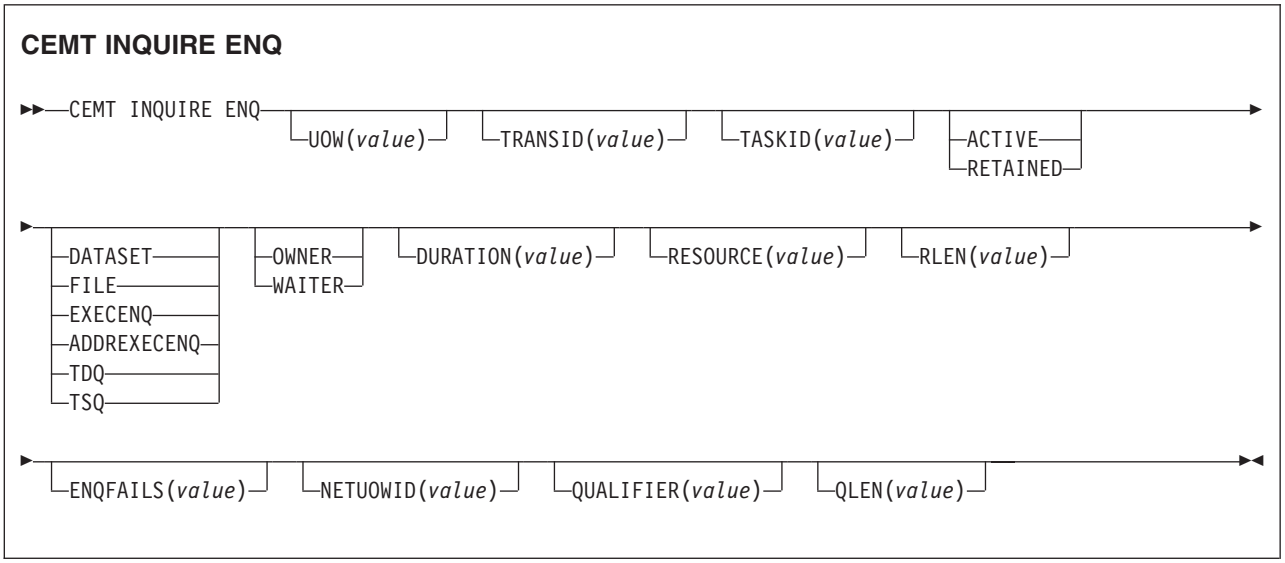
Figure 61. CEMT INQUIRE ENQ screen

Note: When a value does not apply, or is negative (the value begins with No), the fields on the screen are blank. To modify these fields, locate them by tabbing (they appear in the same sequence as in the expanded format), and overtype with input valid for that field. You might find it more convenient to use the expanded format when setting one of these values.

If you place the cursor against a specific entry in the list and press ENTER, CICS displays an expanded format, as shown in Figure 62.

```
INQ ENQ
RESULT
  Enq
  Enq(AC143001D4A72601)
  Transid(CECI)
  Taskid(0000038)
  State(Active)
  Type(Execenq)
  Relation(Owner)
  Duration()
  Resource(ACCOUNTS.DBASE1)
  Rlen(015)
  Enqfails(00000000)
  Netuowid(..GBIBMIYA.IYA2T5C2...Mx...)
  Qualifier()
  Qlen(000)
```

Figure 62. The expanded display of an individual entry



Displayed fields

DURATION (*value*)
Displays the elapsed time (in seconds) since the enqueue entered its current state of OWNER, WAITER or RETAINED.

ENQFAILS (*value*)
Displays, for enqueues held in retained state, the number of failed enqueue attempts for this resource since the enqueue was last acquired. This indicates how many UOWs have received LOCKED responses due to this enqueue. The ENQFAILS option helps you identify which UOWs are causing bottlenecks.

For enqueues held in action state, ENQFAILS displays zero.

ENQSCOPE (*value*)
If the enqueue has sysplex scope, ENQSCOPE displays the 4-character name that was used to qualify the sysplex-wide ENQUEUE request issued by this CICS region. If it has region scope, ENQSCOPE displays blanks.

All CICS systems with the same ENQSCOPE value share the same sysplex Enqueue namespace.

NETUOWID (*value*)
Displays a 27-character string that contains the network-wide LU6.2 name of the UOW that owns, or is waiting for, the enqueue.

Note: Nondisplayable characters appear as periods. You can use PF2 on the expanded panel: to see the value in hexadecimal.

QUALIFIER (*value*)
Displays a 1- through 255-character qualifier (for example, record id) that further identifies the resource associated with the enqueue lock. The data displayed depends on the type of enqueue, as shown in Table 3 on page 208. If no QUALIFIER data applies (that is, for EXECENQ, ADDREXECENQ, or TSQ), a value of zero is displayed.

QLEN (*value*)
Displays the length of the data, in the range 0 through 255, displayed in the QUALIFIER field.

RELATION
Displays whether the data is associated with the owner of the enqueue or with a task waiting for the enqueue. The values are as follows:
OWNER
The (*value*), NETUOWID, TASKID, and TRANSID are those of the owner of the enqueue.
WAITER
The (*value*), NETUOWID, TASKID, and TRANSID are those of a waiter for the enqueue.

RESOURCE (*value*)
Displays a 1- through 255-character string that contains the name of the resource associated with the enqueue lock. The data displayed depends on the type of enqueue, as shown in Table 3 on page 208.

Note: Nondisplayable characters appear as periods. You can use PF2 on the expanded panel: to see the value in hexadecimal.

RLEN (*value*)
Displays the length of the data, in the range 0 through 255, displayed in the RESOURCE field.

STATE
Displays the state that the enqueue is held in. The values are as follows:
ACTIVE
The enqueue is held in active state.
RETAINED
The enqueue is held in retained state. Its owning UOW has been shunted, or is in the process of being shunted.

TASKID (*value*)
Displays the number of the task associated with the UOW. If the UOW is shunted, it is the task number associated with the UOW before it was shunted.

TRANSID (*value*)
Displays the 4-character identifier of the transaction associated with the UOW. If the UOW is shunted, it is the identifier of the transaction associated with the UOW before it was shunted.

TYPE
Displays the type of resource being enqueued upon. The values are as follows:

DATASET
The resource is a record in a VSAM data set (or a CICS-maintained data table). RESOURCE contains the name of the data set and QUALIFIER contains the record identifier.

FILE The resource is a record in either a BDAM file or a user-maintained data table. RESOURCE contains the name of the file and QUALIFIER contains the record identifier.

When the file is a BDAM file, the record identifier is prefixed by the BDAM block identifier. If this combination exceeds 255 characters, truncation occurs.

EXECENQ
The resource is associated with an EXEC CICS ENQ request. RESOURCE contains the enqueue argument passed on the request.

ADDREXECENQ
The resource is associated with an EXEC CICS ENQ request. RESOURCE contains the address enqueue argument passed on the request (that is, the LENGTH parameter was omitted on the request).

TDQ The resource is a logically-recoverable transient data queue. RESOURCE contains the name of the queue. QUALIFIER contains either the string FROMQ or TOQ, indicating whether a read or write lock is held for the queue.

A READQ TD request acquires the FROMQ lock, whereas a WRITEQ TD request acquires the TOQ lock associated with the queue. A DELETEQ TD request acquires both the TOQ and the FROMQ locks.

TSQ The resource is a recoverable temporary storage queue. RESOURCE contains the name of the queue.

The data displayed in the RESOURCE and QUALIFIER fields depends on the resource type, as shown in Table 3.

Table 3. Data displayed in RESOURCE and QUALIFIER

Type	Resource	Qualifier
DATASET	Data set name	Record identifier
EXECENQ	EXEC enqueue argument	None
ADDREXECENQ	Address of EXEC enqueue argument	None
FILE	File name	Record identifier
TDQ	TD queue name	FROMQ or TOQ
TSQ	TS queue name	None

UOW (*value*)
Displays the 16-character local identifier of the associated unit of work.

CEMT INQUIRE ENQMODEL

Retrieve information about enqmodel definitions on the local system.

The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key and type `CEMT INQUIRE ENQMODEL` (or suitable abbreviations for the keywords).

The resulting display is similar to the following example. To change any of the values shown, tab to the highlighted or blank fields and overtype them as required.

Sample screen

```
 INQ ENQMODEL
 STATUS:  RESULTS - OVERTYPE TO MODIFY
  Enqm(*       )                                          NOT FOUND
```

Figure 63. CEMT INQUIRE ENQMODEL screen

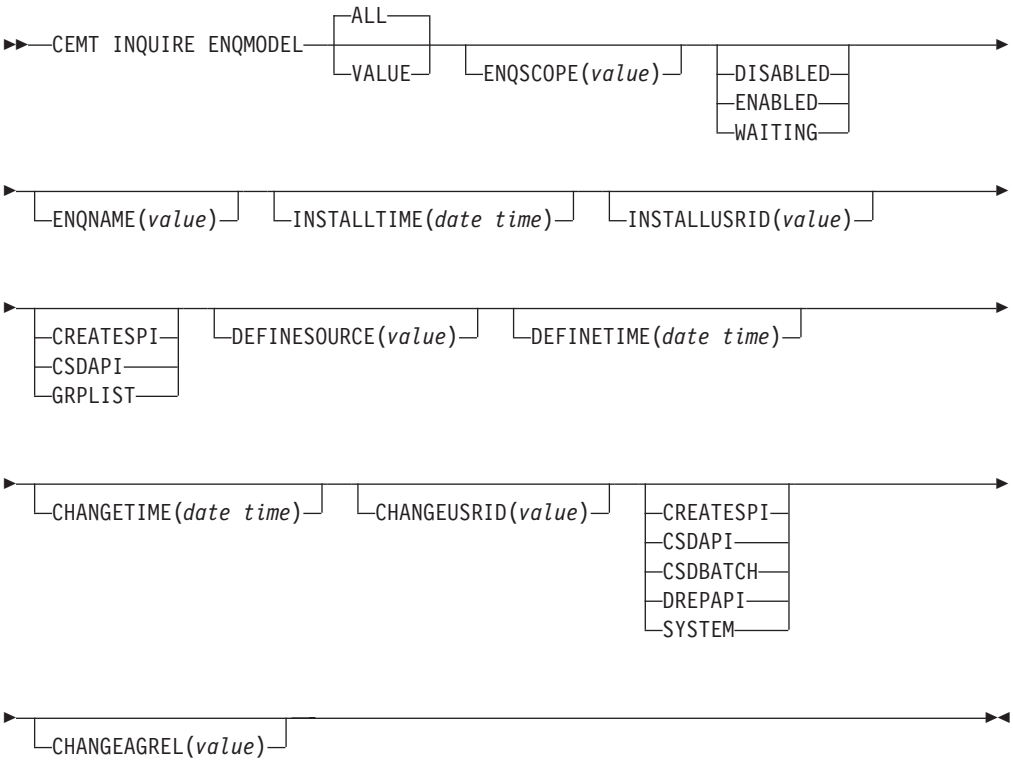
```
i enqm all
STATUS:  RESULTS - OVERTYPE TO MODIFY
 Enqm(fred    ) Enqs(TEST) Ena
    Enqn(HURSLEY                  )
 Enqm(norman  ) Enqs(SYSP) Ena
    Enqn(NORMAN                   )
                                          SYSID=HT61 APPLID=CICSHT61
 RESPONSE: NORMAL                    TIME:  13.23.25  DATE: 02.21.01
PF 1 HELP       3 END       5 VAR       7 SBH 8 SFH 9 MSG 10 SB 11 SF
```

Figure 64. CEMT INQUIRE ENQMODEL expanded display screen

Note: When a value does not apply, or is negative (the value begins with No), the fields on the screen are blank. To modify these fields, locate them by tabbing (they

appear in the same sequence as in the expanded format), and overtype with input valid for that field. You might find it more convenient to use the expanded format when setting one of these values.

CEMT INQUIRE ENQMODEL



Displayed fields

ALL
The default.

CHANGEAGENT(*value*)
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

SYSTEM
The resource definition was last changed by the CICS or CICSPlex system.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

DEFINESOURCE(*value***)**
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

ENQMODEL
The 8-character identifier of the resource definition

ENQNAME
The 1- to 255-character resource or generic name.

EXEC ENQ requests issued by this CICS region are checked against this resource or generic name. If a match is found, and enqscope was specified, the enqueue is to be sysplex-wide, qualified by the 4-character enqscope.

ENQSCOPE
Returns the 4-character name used to qualify SYSPLEX ENQUEUE requests issued by this CICS region. Four blanks indicates that the ENQ is LOCAL.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value*)
Displays the 8-character user ID that installed the resource.

STATUS
Displays the treatment of matching ENQ requests. The values are as follows:
DISABLED
Matching ENQ requests are rejected, and the issuing task abended abcode ENQ_DISABLED. Matching INSTALL/CREATE/DISCARD requests are processed.
ENABLED
Matching ENQ requests are processed in the normal way.
WAITING
Matching ENQ requests are rejected, and the issuing task abended abcode ENQ_DISABLED. There are INSTALL/CREATE/DISCARD requests waiting to be processed.

CEMT INQUIRE EPADAPTER

Retrieve information about an EP adapter.

In the CICS Explorer, the EP adapter operations view provides a functional equivalent to this command.

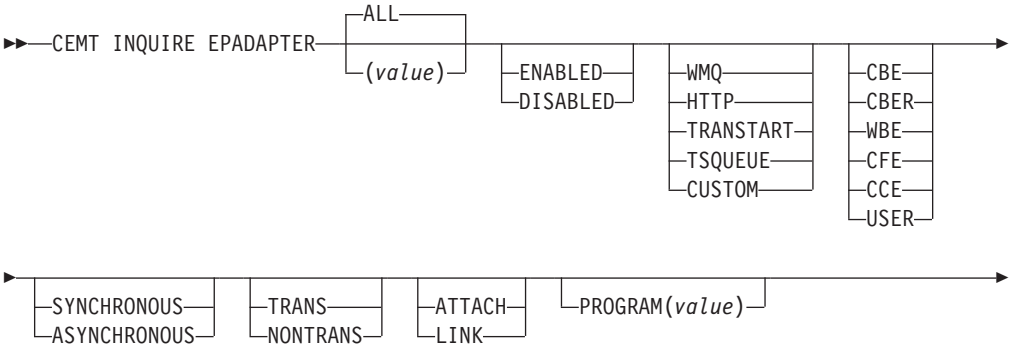
The resource signature

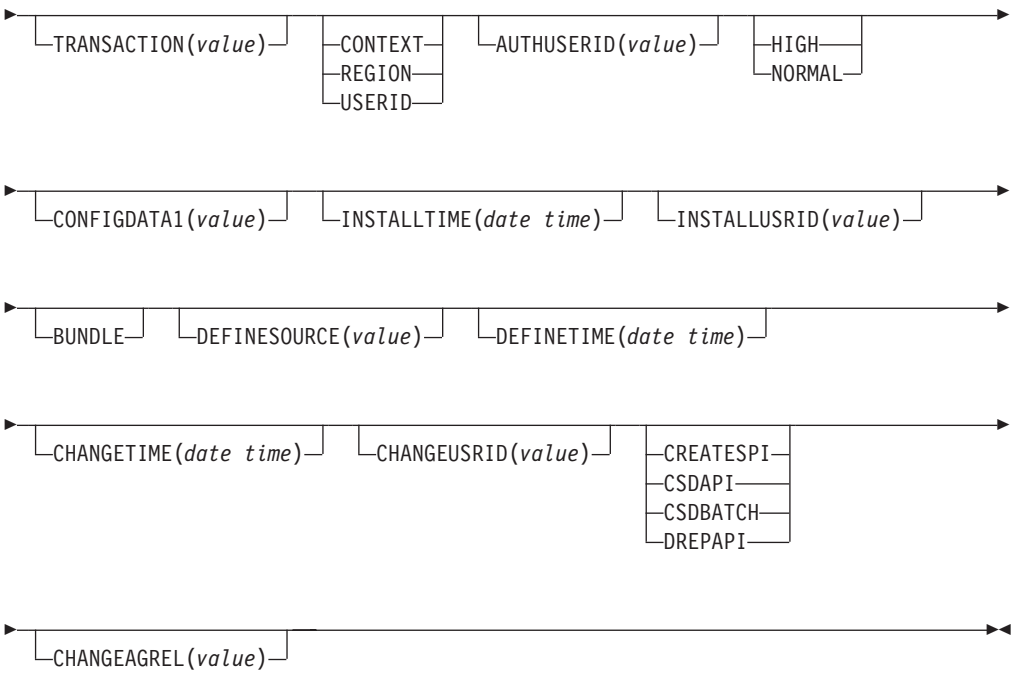
You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key to clear the screen and enter `CEMT INQUIRE EPADAPTER` to obtain information about the EP adapter.

CEMT INQUIRE EPADAPTER





Displayed fields

ADAPTERTYPE
Displays the type of this EP adapter. The possible values are as follows:

CUSTOM
A user-written EP adapter that emits events in any format that you require, to any destination.

HTTP The HTTP EP adapter that emits events to an HTTP server for consumption by products such as WebSphere Business Events and IBM Business Monitor.[1]

TRANSTART
The Transaction Start EP adapter that emits events to a named CICS transaction.

TSQUEUE
The TSQ EP adapter that emits events to a named CICS TS queue.

WMQ The WebSphere MQ EP adapter that emits events to WebSphere MQ for consumption by products such as WebSphere Business Events and IBM Business Monitor.[1]

AUTHORITY
Displays the authority of the EP adapter. The possible values are as follows:

CONTEXT
The EP adapter runs using the user ID of the task that caused the event to be captured. This is always the case when EMITMODE is SYNCHRONOUS or when **Use Context user ID** is specified in the advanced section of the Adapter tab for the EP adapter.

REGION
The EP adapter runs using the CICS region user ID.

USERID
The EP adapter is attached using the identifier specified in the EP adapter user ID and returned in the AUTHUSERID attribute.

AUTHUSERID
Displays the 8-character identifier that is used to attach the EP adapter transaction. This attribute is only set if AUTHORITY is USERID.

CHANGEAGENT(*value***)**
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

CONFIGDATA1
Displays a 64-character data area containing the primary configuration data item for the EP adapter. If the primary configuration data item is less than 64 bytes, the field is padded with blanks. The data item displayed is dependent upon ADAPTERTYPE, as follows:

CUSTOM
Displays the first 64 bytes of custom EP adapter configuration data.

HTTP Displays the 8-character name of the URIMAP definition to be used by an HTTP EP adapter to locate the HTTP server.

TRANSTART
Displays the 4-character name of the event consumer transaction that is started by a Transaction Start EP adapter.

TSQUEUE
Displays the 16-character name of the temporary storage queue for the event emitted by a TSQ EP adapter.

WMQ Displays the 48-character name of the WebSphere MQ queue for event

messages emitted by this WebSphere MQ EP adapter. This data is in the code page defined by the **LOCALCCSID** system initialization parameter.

DATAFORMAT
Displays a CVDA value indicating the format of events emitted by this EP adapter. The possible values are as follows:

CBE Common Base Event format for consumption by products such as IBM Business Monitor.[1]

CBER Common Base Event REST format for the IBM Business Monitor REST HTTP server.[1]

CCE CICS Container Event format.

CFE CICS Flattened Event format.

USER The format is user-defined.

WBE WebSphere Business Events XML format.

DEFINESOURCE(*value***)**
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

EMITMODE(*emitmodevalue***)**
Displays the event emission mode of the event binding:

ASYNCHRONOUS
Event emission is asynchronous of the capturing transaction.

SYNCHRONOUS
Event emission is synchronous with the capturing transaction.

ENABLESTATUS
Displays the status of the event binding:

DISABLED
The event binding is disabled.

ENABLED
The event binding is enabled.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. Only one value is possible:

BUNDLE
The resource was installed by a bundle deployment.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value*)
Displays the 8-character user ID that installed the resource.

INVOKETYPE
Displays a CVDA value indicating how the EP adapter is started. The CVDA values are as follows:

ATTACH
The EP adapter program is attached as a separate task.

LINK The EP adapter program is linked to.

PRIORITY
Displays a CVDA value indicating the dispatching priority of event emission for this EP adapter. This attribute is ignored when EMITMODE is SYNCHRONOUS. The CVDA values are as follows:

HIGH Events emitted for this EP adapter are high priority.

NORMAL
Events emitted for this EP adapter are normal priority.

PROGRAM
Displays the 8-character name of the EP adapter program. If the ADAPTERTYPE is CUSTOM, this attribute is only applicable if INVOKETYPE is LINK.

TRANSACTION
Displays the 4-character name of the transaction definition that is used if the EP adapter transaction is attached. The TRANSACTION attribute is only applicable if INVOKETYPE is ATTACH.

TRANSMODE(*transmodevalue*)
Displays the event transactional mode of the event binding:

NONTRANS
Events are not transactional.

TRANS
Events are transactional.

Note: [1] IBM Business Monitor, formerly known as WebSphere Business Monitor.

CEMT INQUIRE EVENTBINDING

Retrieve information about an event binding.

In the CICS Explorer, the Events bindings operations view provides a functional equivalent to this command.

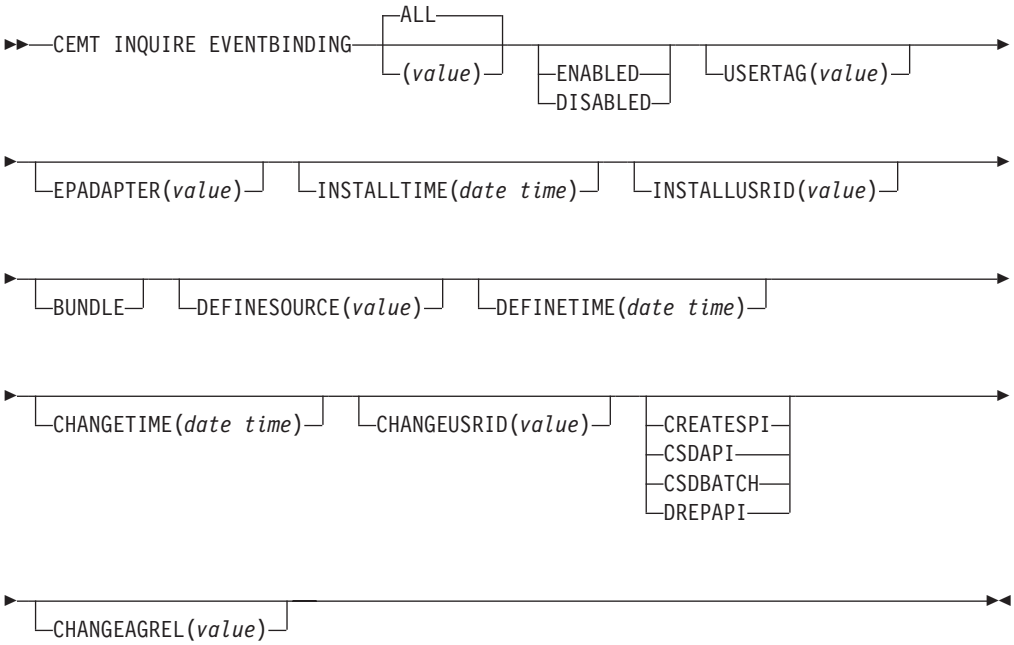
The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key to clear the screen and enter CEMT INQUIRE EVENTBINDING to obtain information about the event binding.

CEMT INQUIRE EVENTBINDING



Displayed fields

CHANGEAGENT(*value*)
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

CHANGEAGREL(*value*)
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time*)
Displays the date and time when the resource definition was last changed. The

format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

DEFINESOURCE(*value***)**
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

ENABLESTATUS
Displays the status of the event binding:

DISABLED
The event binding is disabled.

ENABLED
The event binding is enabled.

EPADAPTER*(value)*
Displays the 32-character name of an EP adapter.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. Only one value is possible:

BUNDLE
The resource was installed by a bundle deployment.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

USERTAG *(value)*
Displays the 8-character user tag of the event binding.

CEMT INQUIRE EVENTPROCESS

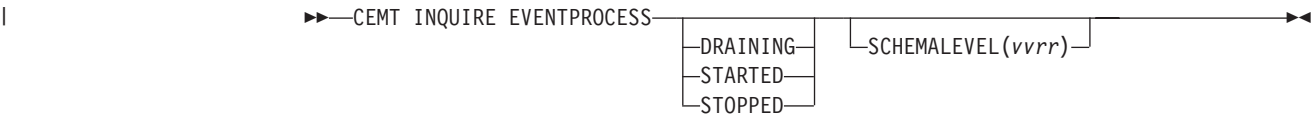
Retrieve the status of event processing.

In the CICS Explorer, the Event processing operations view provides a functional equivalent to this command.

Input

Press the Clear key to clear the screen then enter `CEMT INQUIRE EVENTPROCESS` to retrieve the status of event processing.

CEMT INQUIRE EVENTPROCESS



Displayed fields

EPSTATUS
Displays the status of the event processing:
DRAINING
Event processing is draining.
STARTED
Event processing is started.
STOPPED
Event processing is stopped.

SCHEMALEVEL (*vvrr*)
Returns a 4-character value (*vvrr*) indicating the highest version and release of event binding schema that is supported by CICS, where *vv* is the version and *rr* is the release; for example, 0201 indicates version 2 release 1 of the event binding schema.

CEMT INQUIRE EXCI

Retrieve information about jobs using the external CICS interface.

Description

INQUIRE EXCI identifies the names of batch jobs currently connected to CICS through the interregion communication (IRC) facility. These jobs are using the external CICS interface (EXCI) to link to a server program running in CICS.

Input

Press the Clear key to clear the screen. There are two ways of starting this transaction:

- Type `CEMT INQUIRE EXCI`. You get a display that lists the current status.
- Type `CEMT INQUIRE EXCI` followed by `Task(number)` to limit the scope of the inquiry.

Sample screen

```
INQ EXCI
STATUS:  RESULTS
 Exc(HADBS1..CICSWK1 - MV29            ) Tas(0000121)
 Exc(AHCLN2..TRNSL 1 - MV29            ) Tas(0000125)
    Uri(B 67B1F77EB7D       21 1 1     )
```

Figure 65. CEMT INQUIRE EXCI screen

Note: There are blank fields on the screen where a value does not apply or is 'negative'; that is, the value begins with 'No'. To modify these fields, locate them by tabbing (they appear in the same sequence as in the expanded format), and overtype with input valid for that field. You may find it more convenient to use

the expanded format when setting one of these values, (negating is easier because you are overtyping a displayed field with its negative causing the result to become non-displayed).

If you place the cursor against a specific entry in the list and press ENTER, CICS displays an expanded format as shown in Figure 66.

```
INQ EXCI
RESULT
    Exc(HADBS1..CICSWK1 - MV29           ) Tas(0000121)
    Task(0000121)
```

Figure 66. The expanded display of an individual entry

CEMT INQUIRE EXCI

Displayed fields

EXCI (*value*)
Indicates that this panel relates to an EXCI inquiry.

The value consists of:

jobname.stepname.procname - mvsid

One or both of the *stepname* and *procname* might not be present, indicated by the periods (.) being adjacent to one another.

The *mvsid* identifies the MVS system on which the job is running. If XCF/MRO is in use, the job can reside on a different MVS image from that on which CICS is running.

TASK (*value*)
Displays the task number of the mirror transaction running on behalf of a specific batch job.

Information about jobs using the external CICS interface is available only after that job has issued at least one DPL request. A nonzero task number indicates that a DPL request is currently active. A zero task number indicates that an external CICS interface session is still open (connected) for that job, although no DPL request is currently active.

URID (*value*)
Displays, when the job is using RRMS to coordinate updates, and when there is an active DPL request for the session, a 32-character string containing the hexadecimal representation of the RRMS Unit of Recovery Identifier.

CEMT INQUIRE FEPI

Comments

For information on CEMT INQUIRE FECONNECTION, CEMT INQUIRE FENODE, CEMT INQUIRE FEPOOL, CEMT INQUIRE FEPROPSET, and CEMT INQUIRE FETARGET, see the *CICS Front End Programming Interface User's Guide*.

CEMT INQUIRE FILE

Retrieve information about VSAM, BDAM, and remote files, and files that refer to data tables.

In the CICS Explorer, the Files operations view provides a functional equivalent to this command.

Description

The values that are returned to you can vary depending on when the command is issued. For example, if the file is closed when you issue the command, much of the information you receive is about the state of the file when it is next opened. If the file has never been opened, you receive default or null values for some options, which might change when the file is opened.

The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT INQUIRE FILE` (or suitable abbreviations for the keywords). The resulting display lists the current status of all files installed.
- Type `CEMT INQUIRE FILE` (or suitable abbreviations for the keywords), followed by the attributes that are necessary to limit the range of information that you require. For example, if you enter `cemt i fi v op en up`, the resulting display shows the details of only those files that are VSAM, open, enabled, and can be updated.

You can tab to the highlighted or blank fields and overtype them with the required values.

(*value***)**
Specifies one or more names (1 - 8 characters) of files defined to CICS.

ALL
The default. Information about all files is given, unless you specify a selection of files to be queried.

Sample screen

```
IN FILE
STATUS:  RESULTS - OVERTYPE TO MODIFY
 Fil(DFHCMACD) Vsa Ope Ena Rea                     Sha
       Dsn( CICSTS42.CICS.DFHCMACD                       )
 Fil(DFHCSD  ) Vsa Clo Une Rea Upd Add Bro Del     Sha
       Dsn( CICSTS42.CICS.CICSH.DFHCSD                   )
 Fil(DFHDBFK ) Vsa Clo Ena Rea Upd Add Bro Del     Sha
 Fil(DFHLRQ  ) Vsa Ope Ena Rea Upd Add Bro Del     Sha
       Dsn( CICSTS42.CICS.CICSH.DFHLRQ                   )
 Fil(FILEA   ) Vsa Clo Ena Rea Upd Add Bro Del     Sha
       Dsn( CICSTS42.CICS.CICSH.FILEA                    ) Cft Loa
```

Figure 67. CEMT INQUIRE FILE screen

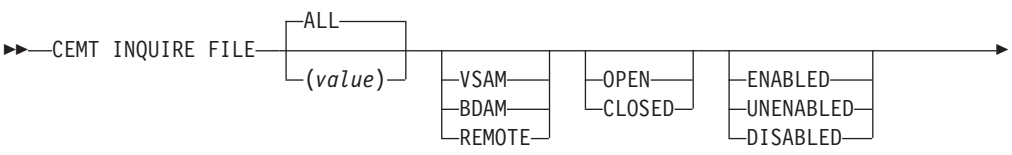
Note: When a value does not apply, or is negative (the value begins with No), the fields on the screen are blank. To modify these fields, locate them by tabbing (they appear in the same sequence as in the expanded format), and overtype with input valid for that field. You might find it more convenient to use the expanded format when setting one of these values.

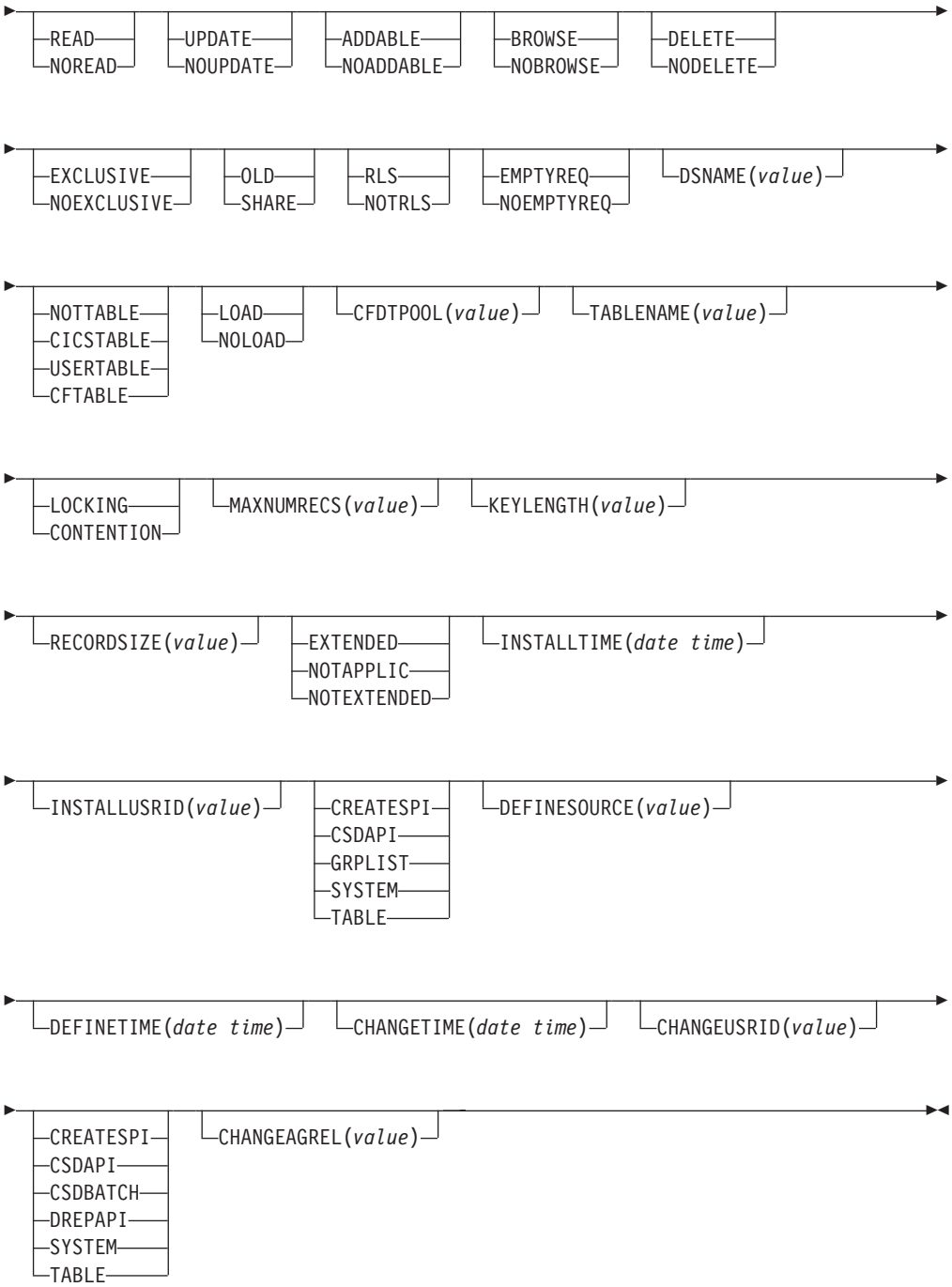
If you place the cursor against a specific entry in the list and press ENTER, CICS displays an expanded format, as shown in Figure 68.

```
 INQ FILE
 RESULT - OVERTYPE TO MODIFY
   File(DFHCSD)
   Accessmethod(Vsam)
   Openstatus( Closed )
   Enablestatus( Unenabled )
   Readstatus( Read )
   Updatestatus( Update )
   Addstatus( Addable )
   Browsestatus( Browse )
   Deletestatus( Delete )
   Exclstatus(              )
   Disposition( Share )
   Rlsaccess( Notrls )
   Emptystatus( Noemptyreq )
   Dsname( CICSTS42.CICS.CICSH.DFHCSD )
   Table( Nottable )
   Loadtype( Noload )
   Cfdtpool(          )
+  Tablename(          )
   Updatemodel(         )
   Maxnumrecs( 00000000 )
   Keylength( 000 )
   Recordsize( 00000 )
   Rbatype( Notapplic )
```

Figure 68. The expanded display of an individual entry

CEMT INQUIRE FILE





Displayed fields

ACCESSMETHOD
Displays the access method of the file.

VSAM
Virtual storage access method (VSAM).

BDAM
Basic direct access method (BDAM).

REMOTE
The file is defined as remote.

ADDSTATUS
Displays whether new records can be added to the file. The values are as follows:

ADDABLE
New records can be added to the file.

NOADDABLE
New records cannot be added to the file.

BROWSESTATUS
Displays whether you can browse the file.

BROWSE
You can browse records in the file.

NOBROWSE
You cannot browse records in the file.

CFDTPOOL (*name***)**
Displays, for a file that refers to a coupling facility data table, the 8-character name of the coupling facility data table pool in which the CFDT resides.

If the pool name is not defined, the name is displayed as eight blanks. If the pool name is specified, it is always displayed even if the file definition does not refer to a coupling facility data table.

CHANGEAGENT(*value***)**
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

SYSTEM
The resource definition was last changed by the CICS or CICSPlex system.

TABLE
The resource definition was last changed by a File Control table definition.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value*)
Displays the 8-character user ID that ran the change agent.

DEFINESOURCE(*value*)
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time*)
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

DELETESTATUS
Displays whether you can delete records from the file.

DELETE
You can delete records from the file.

NODELETE
You cannot delete records from the file.

DISPOSITION
Displays the disposition of this file.

OLD The disposition of the data set associated with the file is defined as unshared for dynamic allocation.

SHARE
The disposition of the data set associated with the file is defined as shared for dynamic allocation.

DSNAME (*value*)
Displays the name (1- 44 characters) of the VSAM or BDAM data set with which this file is associated.

For a coupling facility data table, CICS displays the name of the source data set if the file specifies that the table is initially loaded from a data set; otherwise, CICS displays blanks.

CICS also displays blanks for files that have their DSNAME specified on a DD statement in the CICS JCL if the file has not been opened since the last cold or initial start.

EMPTYSTATUS (*VSAM only*)
Displays whether the data set is emptied when a file that references it is next opened. This field is valid only for data sets that are defined as reusable and are accessed in non-RLS mode. The values are as follows:

EMPTYREQ
Applies only to a VSAM file that is defined to VSAM as a cluster with the REUSE parameter. It indicates that when a file is next opened, its data is erased.

NOEMPTYREQ
Applies only to VSAM files. It indicates that when a file is next opened, its data is not erased.

ENABLESTATUS
Displays whether transactions can access the file.

ENABLED
The file is available for use by transactions and, if closed, it is opened on the first request.

UNENABLED
The file is not available for use by transactions except for those that are currently using it. If there are any such users, BEING CLOSED is also displayed.

This status is the same as DISABLED except that it occurs implicitly when a SET FILE CLOSE is requested. The file is enabled implicitly by a SET FILE OPEN command.

DISABLED
The file is not available for use by transactions except for those that are currently using it. If there are any such users, BEING DISABLED is also displayed.

The file can be reenabled by a SET FILE ENABLED command. See also UNENABLED.

You cannot disable a remote file.

EXCLSTATUS (*BDAM only***)**
Displays whether records on this file are placed under exclusive control when a read for update is issued.

EXCLUSIVE
Records on the file are protected from simultaneous update.

NOEXCLUSIVE
Records on the file are not protected from simultaneous update.

FILE (*value***)**
Indicates that this panel relates to a file inquiry and displays an 8-character file name.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

SYSTEM
The resource was installed by the CICS or CICSPlex SM system.

TABLE
The resource was installed by using a File Control table definition.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

KEYLENGTH (*value***)**
Displays the length of the record key for a file associated with a VSAM KSDS

or a file associated with a coupling facility data table. If the file is associated with a BDAM data set, the value is the length of the logical key used for deblocking.

Note:

1. If the file is closed and the key length is not defined in the file definition, CICS displays a zero value.
2. If the file is closed and a key length is defined on the file definition, CICS displays the value from the file definition.
3. Generally, if a file is open, CICS obtains the key length from the associated data set, in which case CICS displays the value from the data set. However, files that refer to coupling facility data tables defined with LOAD(NO) must get their key length from the file definition, in which case CICS returns the defined value when the file is open. This value must also match that of the coupling facility data table if it has already been created.

LOADTYPE

Displays the load type for a coupling facility data table.

LOAD

The coupling facility data table is, or will be, preloaded from a source data set.

NOLOAD

The coupling facility data table is not preloaded from a source data set.

MAXNUMRECS (*value*)

Displays the maximum number of records that the data table for this file can hold.

Note:

1. If the file resource definition specifies a MAXNUMRECS numeric value, even though the object is not a table (NOTTABLE is displayed on the TABLE option), CICS displays the specified value. The value can be in the range 1 - 99999999.
2. If the file resource definition is specified with MAXNUMRECS(NOLIMIT), meaning the number of records is unlimited, CICS displays a value of zero. (Internally, CICS holds NOLIMIT as the maximum positive fullword value (+2147483647 or X'7FFFFFFF').
3. If the file is remote, CICS displays a value of minus 1 (-1).
4. If the object is a coupling facility data table, note these points:
   - The maximum number of records can be altered by a coupling facility data table server command, leaving the file definition MAXNUMRECS value unchanged. CICS displays the value in the file definition until the file is opened, after which CICS displays the MAXNUMRECS value defined to the server.
   - If the value is changed again by a coupling facility data table server command, CICS obtains and displays the new value only after a file control operation on the table. Until then, CICS continues to display the old value.
   - You can use the server DISPLAY TABLE console command to obtain the current value for a coupling facility data table.

Note: You can reset this value by overtyping it with a different value. The new value takes effect only when a new data table is created (which happens on

each file open for a CICS-maintained or user-maintained data table, but only when the table does not already exist for a CF data table).

OPENSTATUS
Displays the open status of the file.

OPEN
The file is open. If the file is OPEN ENABLED, it is available for data accesses by CICS transactions. If it is OPEN DISABLED, it must first be enabled before it is available.

CLOSED
The file is closed; the access method does not allow accesses to the data until the file is opened. The file can be opened either explicitly by the user or, if the file has the ENABLED attribute, implicitly by CICS on the next reference to the file.

A file with the DISABLED attribute can be enabled only by a SET FILE ENABLED command. A file with the UNENABLED attribute can be enabled by a SET FILE ENABLED command or SET FILE OPEN command.

The CLOSED status might be a temporary status for a file that has just come out of LOADMODE status.

You can reset this value by overtyping it with one of the above OPENSTATUS values, or with:

FORCECLOSE
Specifies that the file is to be closed, and the tasks of any existing users of the file are to be stopped.

RBATYPE(*value*)
Displays whether, for VSAM files, the data set uses extended addressing.

EXTENDED
This VSAM data set uses extended relative byte addressing and therefore can hold more than 4 gigabytes of data.

NOTAPPLIC
One of the following is true:
- The data set is BDAM.
- The file is remote.
- The file is not open.

NOTEXTENDED
This VSAM data set does not use extended relative byte addressing and therefore cannot hold more than 4 gigabytes of data.

You cannot modify the contents of this field.

READSTATUS
Displays whether you can read records from the file.

READ
You can read records in the file.

NOREAD
You cannot read records in the file.

RECORDSIZE (*value*)
Displays the actual size of fixed-length records, or the maximum size of variable-length records.

If the file is not open, CICS displays the value specified in the installed file definition. If the file definition does not specify a record size, CICS displays a zero value.

Generally, if a file is open, CICS obtains the record size from the associated data set, in which case CICS displays the value from the data set. However, files that refer to coupling facility data tables defined with LOAD(NO) must get their record size from the file definition, in which case CICS returns the defined value when the file is open. This value must also match that of the coupling facility data table if it has already been created.

RLSACCESS (*VSAM only*)
Displays whether the file is defined to be accessed in RLS mode.
RLS The file is defined to be accessed in RLS mode.
NOTRLS
The file is defined to be accessed in a non-RLS mode (LSR or NSR)

TABLE
Displays whether the file name specified on the FILE parameter represents a data table.
NOTTABLE
The file name does not represent a data table.
CFTABLE
The file name represents a coupling facility data table.
CICSTABLE
The file name represents a CICS-maintained data table.
USERTABLE
The file name represents a user-maintained data table.

TABLENAME (*value*)
Displays the table name specified for the coupling facility data table on the file resource definition, if one is specified, or displays the file name if no table name was specified on the file resource definition.

CICS displays blanks if the file does not refer to a coupling facility data table

UPDATEMODEL (CFDT ONLY)
Displays the update model specified for the coupling facility data table in the installed file definition.
CONTENTION
The coupling facility data table is updated using the contention model.
LOCKING
The coupling facility data table is updated using the locking model.

UPDATESTATUS
Displays whether the file is updatable.
UPDATE
You can update records in the file.
NOUPDATE
You cannot update records in the file.

CEMT INQUIRE GTFTRACE

Show the MVS Generalized Tracing Facility (GTF) status.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

The **CEMT INQUIRE GTFTRACE** command shows whether CICS tracing to the MVS Generalized Tracing Facility (GTF) is started or stopped.

This command is intended for use at terminals that do not support the CETR full-screen tracing transaction.

For background information about GTF tracing, see the *CICS Problem Determination Guide*.

Input

Press the Clear key and type `CEMT INQUIRE GTFTRACE` (or suitable abbreviations for the keywords).

The resulting display is similar to the following example. To change the displayed value, tab to the highlighted field and overtype it as required.

Sample screen

```
IN GTFTRA
STATUS:  RESULTS - OVERTYPE TO MODIFY
 Gtf Sto
```

Figure 69. CEMT INQUIRE GTFTRACE screen

If you place the cursor against the result and press ENTER, CICS displays an expanded format, as shown in Figure 70.

```
IN GTFT
RESULT - OVERTYPE TO MODIFY
  Gtftrace
  Status( Stop )
```

Figure 70. The expanded display

CEMT INQUIRE GTFTRACE

Displayed fields

GTFTRACE
Indicates that this panel relates to a GTFTRACE inquiry.

STATUS
Displays the status of CICS tracing to the MVS Generalized Tracing Facility (GTF). The values are:

START
CICS tracing to the MVS Generalized Tracing Facility (GTF) is in progress.

STOP
CICS tracing to the MVS Generalized Tracing Facility (GTF) is not in progress.

CEMT INQUIRE HOST

Inquire on the status of a virtual host.

Description

CEMT INQUIRE HOST allows you to retrieve information about a particular virtual host in the local CICS region.

Virtual hosts are based on the URIMAP resource definition object. CICS automatically creates virtual hosts for you by grouping all the URIMAP definitions in a CICS region that specify the same host name and the same TCPIPSERVICE into a single data structure. URIMAP definitions that do not specify a TCPIPSERVICE are added to all the data structures that specify a matching host name, so these URIMAP definitions might be part of more than one data structure.

Syntax

Press the Clear key and type `CEMT INQUIRE HOST` to display a list of hosts. If you want to change the displayed value, you can tab to the highlighted field and over type it as required.

Sample screen

```
 IN HOST
 STATUS:  RESULTS - OVERTYPE TO MODIFY
   Hos(www.example.com  )              Ena
   Hos(www.example.org  )              Ena
```

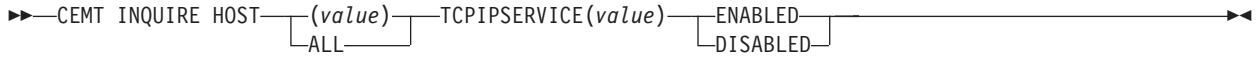
Figure 71. CEMT INQUIRE HOST screen

If you place the cursor against the result and press Enter, CICS displays an expanded format as shown in Figure 72.

```
IN HOST
RESULT - OVERTYPE TO MODIFY
    Host(www.example.com)
    Tcpipservice(TEST2)
    Enablestatus( Enabled )
```

Figure 72. The expanded display

CEMT INQUIRE HOST



Options

(*value***)**
Specifies the name of a virtual host. The name of each virtual host is taken from the host name specified in the URIMAP definitions that make up the virtual host. For example, if your CICS region contains URIMAP definitions that specify a host name of `www.example.com`, CICS creates a virtual host with the same name. A host name in a URIMAP definition can be up to 120 characters.

TCPIPSERVICE *(value)*
Displays the name of the TCPIPSERVICE definition that specifies the inbound port to which this virtual host relates. If this definition is not given, the virtual host relates to all TCPIPSERVICE definitions.

ENABLESTATUS
Displays the status of the virtual host. The possible values are:

ENABLED
The URIMAP definitions that make up the virtual host can be accessed.

DISABLED
The URIMAP definitions that make up the virtual host cannot be accessed.

CEMT INQUIRE INTTRACE

Show the status of CICS internal tracing.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

The **INQUIRE INTTRACE** command shows whether CICS internal tracing is in progress.

This command is intended for use at terminals that do not support the full-screen tracing transaction CETR.

Input

Press the Clear key and type `CEMT INQUIRE INTTRACE` (or suitable abbreviations for the keywords).

The resulting screen is similar to the following example. To change the displayed value, tab to the highlighted field and overtype it as required.

Sample screen

```
IN INTTRA
STATUS:  RESULTS - OVERTYPE TO MODIFY
 Int Sta
```

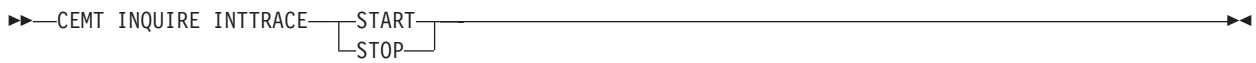
Figure 73. CEMT INQUIRE INTTRACE screen

If you place the cursor against the result and press ENTER, CICS displays an expanded format, as shown in Figure 74.

```
IN INTT
RESULT - OVERTYPE TO MODIFY
  Inttrace
  Status( Start )
```

Figure 74. The expanded display

CEMT INQUIRE INTTRACE



Displayed fields

INTTRACE
Indicates that this panel relates to an INTTRACE inquiry.

STATUS
Displays the status of CICS tracing to the MVS Generalized Tracing Facility (GTF). The values are:
START
CICS internal tracing is in progress.
STOP
CICS internal tracing is not in progress.

CEMT INQUIRE IPCONN

CEMT INQUIRE IPCONN returns information about the status of IPIC connections to a remote system or to another CICS region.

In the CICS Explorer, the IPIC connections operations view provides a functional equivalent to this command.

Description

An IPCONN (IPIC connection) is a TCP/IP communication link from your local CICS region to another CICS region, or another system. See for information about MRO and ISC over SNA connections.

You can reset the options either by typing the SET command or by overtyping at the appropriate place on the INQUIRE screen.

If you want to install a new IPCONN definition when one is already installed, you must set the connection OUTSERVICE before using the CEDA INSTALL command for your new definition.

For information about the different kinds of intercommunication connections, see the *CICS Intercommunication Guide*.

The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key to clear the screen. You can start this transaction in two ways:

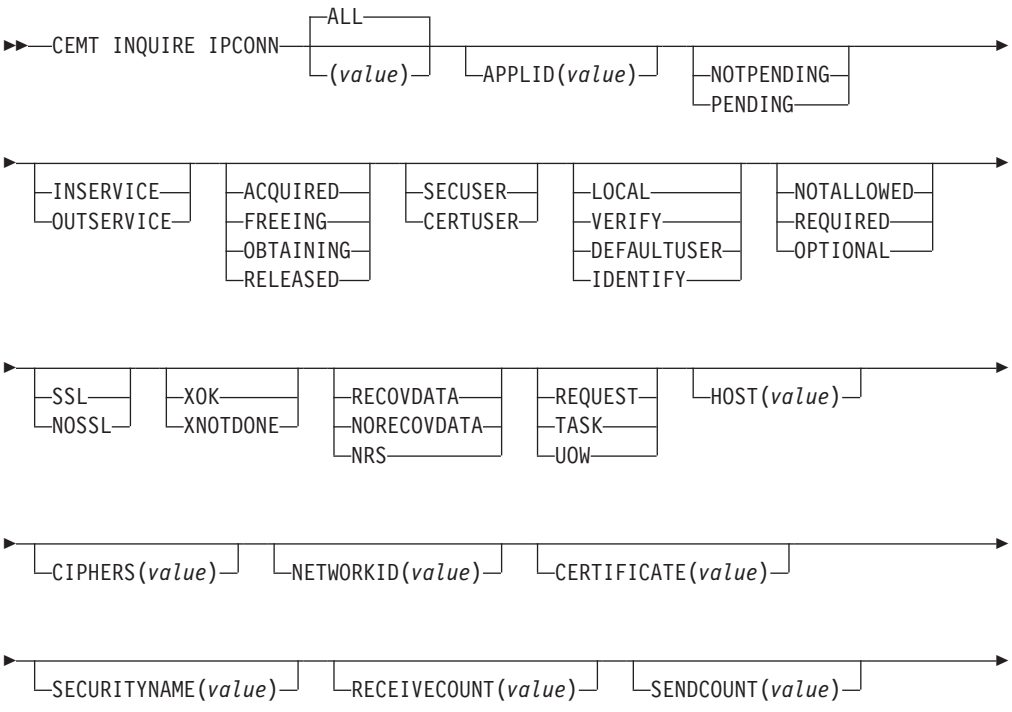
- Type `CEMT INQUIRE IPCONN` (or suitable abbreviations for the keywords). The resulting display lists the current status.
- Type `CEMT INQUIRE IPCONN` (or suitable abbreviations for the keywords), followed by the attributes that you need to limit the range of information that you require.

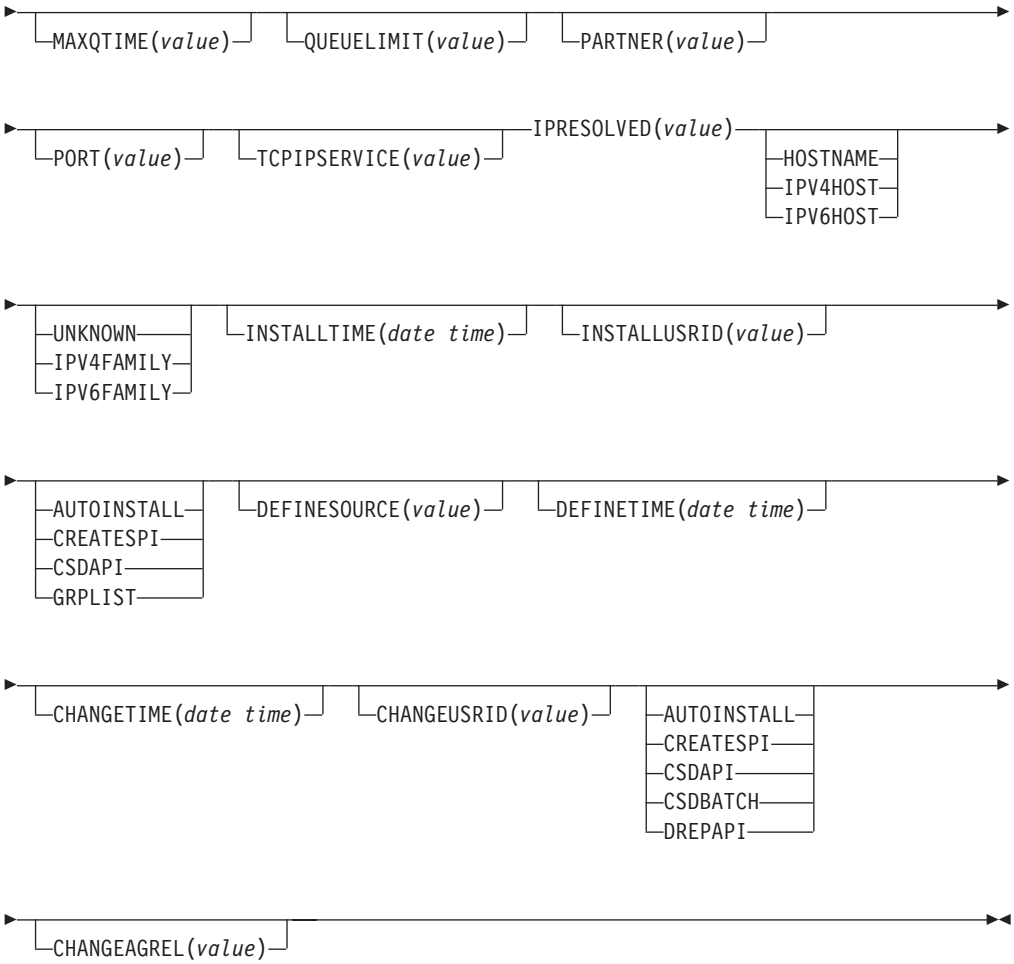
You can tab to the highlighted or blank fields and overtype them with the required values.

(*value***)**
Specifies one or more names (1 - 8 characters) defined for an IPIC connection.

ALL
Is the default.

CEMT INQUIRE IPCONN





Displayed fields

APPLID(*value*)
Displays the name by which the remote system is known to the network, taken from the APPLID option of the IPCONN definition. This value is the application identifier (*applid*) of the remote system, as specified on the APPLID option of its system initialization parameter. For XRF systems, it is the generic applid.

CERTIFICATE(*value*)
Displays a 32-character area containing the label of the certificate, in the key ring, that is used as a client certificate in the SSL handshake for outbound IPCONN connections. If the label is blank, the certificate nominated as the default for the key ring is used.

CHANGEAGENT(*value*)
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

AUTOINSTALL
The resource was autoinstalled.

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

CIPHERS(*value***)**
Displays a 56-character area containing the list of cipher suites that is used to negotiate with clients during the SSL handshake. The list is set by the ENCRYPTION system initialization parameter, but you can edit the list to remove or change the order of cipher suites. See the *CICS RACF Security Guide*.

CONNSTATUS(*value***)**
Indicates the state of the IPCONN between CICS and the remote system. The possible values are as follows:

ACQUIRED
The IPCONN is acquired. The criterion for ACQUIRED is that the capabilities exchange is complete.

FREEING
The IPCONN is being released.

OBTAINING
The IPCONN is being acquired. The IPCONN remains in the OBTAINING state until the criterion for ACQUIRED has been met.

RELEASED
The IPCONN is RELEASED. Although it might also be in INSERVICE status, it is not usable.

Any of the following general conditions can cause the RELEASED status:
- The remote system has not yet initialized.
- No IPCONN definition exists on the remote system and autoinstall was not active or not successful.
- The IPCONN definition on the remote system has been set out of service.
- AUTOCONNECT(NO) has been specified on the IPCONN definition.
- The IPIC connection had been acquired but has since been released by an explicit operator command.

DEFINESOURCE(*value*)
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time*)
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

HOST(*value*)
Displays the host name, IPv4, or IPv6 address of the remote system. All IPv4 addresses are displayed as native IPv4 dotted decimal addresses, for example, `1.2.3.4`, regardless of which type of address format is used. For information about accepted IPv4 formats, see the *CICS Internet Guide*.

HOSTTYPE(*value*)
Displays the address format of the HOST option. The HOSTTYPE option is available using an expanded query only. The possible values are as follows:

HOSTNAME
The HOST option contains a host name.

IPV4HOST
The address is an IPv4 address.

IPV6HOST
The address is an IPv6 address.

IDPROP(*value*)
Displays whether the sender includes the distributed identity in requests over the IPIC connection. The IDPROP option is meaningful only if a connection extends outside a sysplex and is used primarily to prevent distributed identities being transmitted between enterprises. If the connection is between systems in the same sysplex, the value returned by this option is ignored, and the connection operates as if IDPROP(OPTIONAL) is specified.

The possible values are as follows:

NOTALLOWED
A user ID associated with the sending transaction is sent for requests using this connection. NOTALLOWED is the default value.

OPTIONAL
A distributed identity is sent, if available. The user ID associated with the sending transaction is also sent.

REQUIRED
A distributed identity is required for requests using this connection. If REQUIRED is specified, the receiving system must support distributed identities. The user ID associated with the sending transaction is not sent.

INSTALLAGENT(*value*)
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

AUTOINSTALL
The resource was autoinstalled.

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

IPFAMILY(*value***)**
Displays the address format of the IPRESOLVED option. The IPFAMILY option is available using an expanded query only. Filtering interacts with the IPFAMILY option when you filter using wildcard characters. For example, if the value in IPFAMILY is IPv6FAMILY, you must use the colon character to retrieve an IP address. The possible values are as follows:

IPV4FAMILY
The address is an IPv4 address.

IPV6FAMILY
The address is an IPv6 address.

UNKNOWN
IPRESOLVED is not yet in use, or the address cannot be resolved.

IPRESOLVED(*value***)**
Displays the IPv4 or IPv6 address of the host. The IPRESOLVED option is available using an expanded query only.

LINKAUTH(*value***)**
Indicates how the user ID for link security is established in a CICS system with security initialized (SEC=YES).

CERTUSER
TCP/IP communication with the partner system must be configured for SSL and a certificate must be received from the partner system during SSL handshake.

The IPCONN must refer to a TCPIPSERVICE that is defined with SSL(CLIENTAUTH).

The received certificate must be defined to the external security manager so that it is associated with a user ID, which is used to establish link security.

SECUSER
Specifies that the user ID specified in SECURITYNAME is used to establish link security.

MAXQTIME(*value***)**
Displays the maximum time, in seconds, for which allocate requests can be queued. The value is in the range 0 - 9999 or has the standard null value of -1 if MAXQTIME(NO) is specified on the IPCONN definition.

MIRRORLIFE(*value*)
Indicates the minimum lifetime of the mirror task for function-shipped file control, transient data, and temporary storage requests received by this region. This parameter only takes affect when it is specified on the IPCONN definition on the resource-owning region. It is not valid for function shipping interval control or link requests. The possible values are as follows:

REQUEST
The mirror task terminates as soon as possible.

TASK The mirror task remains available to the application that issues the remote request the until the application task ends.

UOW The mirror transaction remains available to the application that issues the remote request until the next sync point is issued.

NETWORKID(*value*)
Displays the network ID of the remote system. The value returned is an 8-byte character string, which is the value of the NETWORKID option of the IPCONN definition. If NETWORKID is not specified on the IPCONN definition, the value returned is the z/OS Communications Server NETID or, for z/OS Communications Server VTAM=NO systems, the value of the UOWNETQL system initialization parameter of this CICS; that is, the CICS on which the IPCONN definition is installed.

The NETWORKID value is used in combination with the APPLID value to ensure unique naming for connecting systems.

PARTNER(*value*)
Displays a 64-character string indicating the product token of the partner system. The field is blank when the connection is not acquired or if the partner system does not indicate a product type when the connection is established. For example, the partner system is `IBM_CICS_Transaction_Server/4.1.0(zOS)` for a CICS TS 4.1 partner.

PENDSTATUS(*value*)
Displays whether pending units of work exist for this IPIC connection. The possible values are as follows:

NOTPENDING
No mismatch of lognames has occurred with the partner.

PENDING
Resynchronization work is outstanding for the connection but the partner system has performed an initial start, preventing completion of the resynchronization process. You can use the SET IPCONN NOTPENDING command to unilaterally commit or back out the units of work associated with the connection, according to their associated transaction definitions. You can also investigate the units of work individually and force them to commit or back out, in which case you must also complete the recovery activity by using a SET IPCONN NOTPENDING command to clear the PENDING condition.

If this is a CICS-to-CICS IPIC connection, no new sync point work (that is, work involving sync level 2 protocols) can be transmitted across the connection until a SET IPCONN NOTPENDING command has been issued.

If you are not concerned by the loss of synchronization caused by the initial (or cold) start of the partner, you can cause the SET IPCONN NOTPENDING command to be issued automatically by specifying XLNACTION(FORCE) on the IPCONN definition.

For further information about pending units of work, see the *CICS Intercommunication Guide*.

PORT(*value***)**
Displays the port number to be used for outbound requests on this connection; that is, the number of the port on which the remote system will be listening. This value might be set to NOTAPPLIC if the connection is never used for outbound traffic, as is usually the case for autoinstalled IPCONN resources.

QUEUELIMIT(*value***)**
Displays the maximum number of allocate requests that can be queued for this connection. The value is in the range 0 - 9999 or has the standard null value of -1 if QUEUELIMIT(NO) is specified on the IPCONN definition.

RECEIVECOUNT(*value***)**
Displays the number of RECEIVE sessions defined for this connection.

RECOVSTATUS(*value***)**
Indicates whether resynchronization work is outstanding for the IPIC connection. The IPIC connection might never have been connected, it might have been quiesced and all resynchronization work completed, or it might have been disrupted without quiesce, in which case resynchronization might be necessary. The possible values are as follows:

NORECOVDATA
Neither side has recovery information outstanding.

NRS CICS does not have recovery outstanding for the IPIC connection, but the partner might have.

RECOVDATA
Indoubt units of work are associated with the IPIC connection, or outstanding resynchronization tasks are awaiting FORGET on the connection. Resynchronization takes place when the IPIC connection next becomes active or when the UOW is unshunted.

If recovery is outstanding, on completion of exchange of lognames either resynchronization takes place or, in the case of a cold exchange, the PENDING condition is created.

SECURITYNAME(*value***)**
Displays the security name of the remote system.

In a CICS system with security initialized (SEC=YES), the security name is used to establish the authority of the remote system.

The security name must be a valid RACF® user ID on this region. Access to protected resources on this region is based on the RACF user profile and its group membership.

SENDCOUNT(*value***)**
Displays the number of SEND sessions defined for this IPIC connection.

SERVSTATUS(*value***)**
Indicates whether data can be sent and received on the IPIC connection. The possible values are as follows:

INSERVICE
Data can be sent and received.

OUTSERVICE
Data cannot be sent or received.

SSLTYPE(*value***)**
Displays the level of secure sockets support that is being used for this service.

NOSSL
The Secure Sockets Layer is not being used for this service.

SSL The Secure Sockets Layer is being used for this service.

TCPIPSERVICE(*value*)
Displays the name of a PROTOCOL(IPIC) TCPIPSERVICE definition that defines the attributes of the inbound processing for this IPIC connection.

USERAUTH(*value*)
Indicates how the user ID for attach-time user security is established in a CICS system with security initialized (SEC=YES). The possible values are as follows:

DEFAULTUSER
CICS does not accept a user ID and password from the partner system. All requests run under the default user ID.

IDENTIFY
Incoming attach requests must specify a user ID but not a password.

LOCAL
CICS does not require a user ID or password from clients. All requests will run under the link user ID.

VERIFY
Incoming attach requests must specify a user ID and password.

CEMT INQUIRE IRBATCH

Retrieve information about jobs using the external CICS interface.

Description

CEMT INQUIRE IRBATCH is retained for compatibility. You should use CEMT INQUIRE EXCI.

CEMT INQUIRE IRC

Show the status of the interregion communication (IRC) facility.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

The **INQUIRE IRC** command shows whether the interregion communication (IRC) facility is open or closed.

Input

Press the Clear key and type `CEMT INQUIRE IRC` (or suitable abbreviations for the keywords).

The resulting display is similar to the following example.

Sample screen

```
IN IRC
STATUS:  RESULTS - OVERTYPE TO MODIFY
 Irc Clo DFHIR000
```

Figure 75. CEMT INQUIRE IRC screen

If you place the cursor against the result and press ENTER, CICS displays an expanded format, as shown in Figure 76.

```
IN IRC
RESULT - OVERTYPE TO MODIFY
  Irc
  Openstatus( Closed )
  Xcfgroup( DFHIR000 )
  Return(00000000)
  Reason(00000000)
```

Figure 76. The expanded display

CEMT INQUIRE IRC

Displayed fields

IRC
Indicates that this panel relates to an IRC inquiry.

OPENSTATUS (*value*)
Displays whether the IRC facility is open or closed. The values are:

CLOSED
Interregion communication for this CICS system is terminated after tasks using the session have terminated normally.

OPEN
Interregion communication for this CICS system is initialized. This allows connections to be established between this system and other systems. You can control the availability of individual connections with the INSERVICE or OUTSERVICE settings on the connection names, using CEMT INQUIRE CONNECTION or CEMT SET CONNECTION.

The DFHIR3780 message is issued to the console log if the region made an unsuccessful attempt to start IRC.

Note:

1. You can reset this value by overtyping it with a different value, or with:
   IMMCLOSE
   All tasks using IRC sessions are to be abended when they next try to use the session. Interregion communication is terminated when all tasks that were using IRC have been abended or have terminated normally.

2. The values of Return and Reason will be nonzero only after certain SET (or overtype) failures. These are accompanied by a message number, which you can refer to in *CICS Messages and Codes Vol 1* .

XCFGROUP
Displays the name of the cross-system coupling facility (XCF) group of which this region is a member.

If this region is not a member of an XCF group (because it has not signed on to IRC), XCFGROUP displays the XCF group the region would be in if XCF were opened.

For introductory information about XCF/MRO, see the *CICS Intercommunication Guide*.

CEMT INQUIRE JMODEL

Retrieve information about journal models and log stream names.

Description

The INQUIRE JMODEL command enables you to inquire on installed journal models to obtain corresponding log stream names.

The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the CLEAR key to clear the screen. There are two ways of starting this transaction:

- Type `CEMT INQUIRE JMODEL` (the minimum abbreviation is `CEMT I JM`). You get a display that lists the current status.
- Type `CEMT INQUIRE JMODEL` (`CEMT I JM`) followed by as many of the other attributes as are necessary to limit the range of information that you require. So, for example, if you enter `cemt i jm sm`, the resulting display will show you the details of only those journal models that are of SMF log stream type.

(*value***)**
specifies one or more journal models.

ALL
is the default.

Sample screen

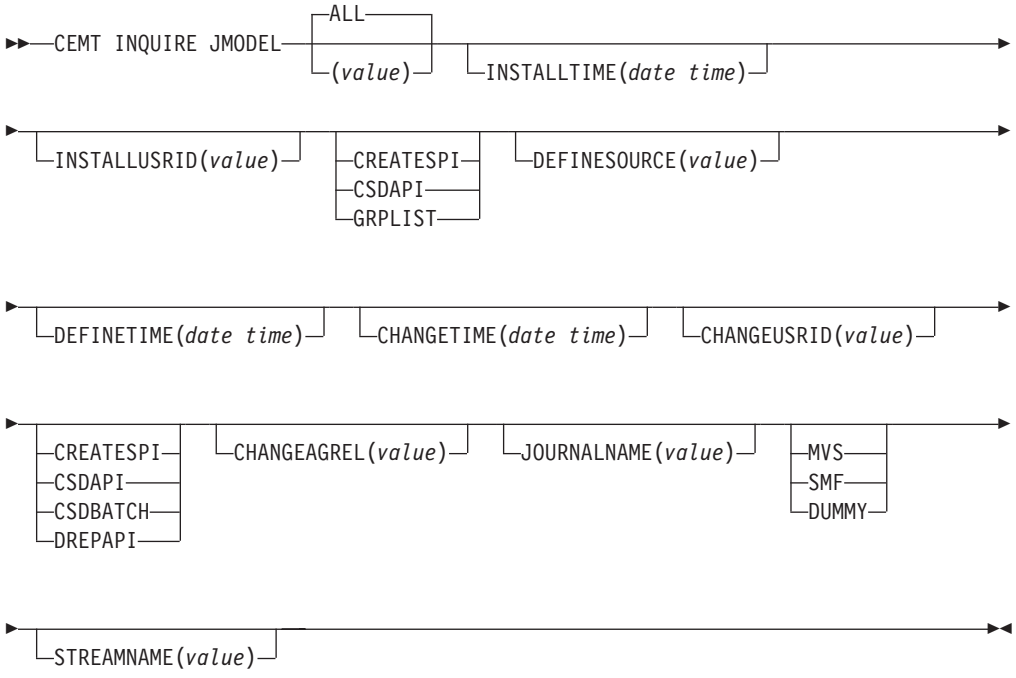
```
IN JMODE
STATUS:  RESULTS
 Jmo(DFHJ02  ) Jou(DFHJ02  ) Mvs
    Str(BELL.TESTLOG.FWDRECOV     )
 Jmo(DFHLOG  ) Jou(DFHLOG  ) Mvs
    Str(&USERID..&APPLID..&JNAME. )
 Jmo(DFHSHUNT) Jou(DFHSHUNT) Mvs
    Str(&USERID..&APPLID..&JNAME. )
```

If you place the cursor against a specific entry in the list and press ENTER, CICS displays an expanded format as shown in Figure 77.

```
IN JMODE
RESULT
  Jmodel(DFHJ02)
  Journalname(DFHJ02)
  Type(Mvs)
  Streamname(BELL.TESTLOG.FWDRECOV)
```

Figure 77. The expanded display of an individual entry

CEMT INQUIRE JMODEL



Displayed fields

JMODEL (*value***)**
indicates that this panel relates to a journal model inquiry and displays the 8–character name of an installed journal model.

CHANGEAGENT(*value***)**
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

DEFINESOURCE(*value***)**
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

JOURNALNAME
displays the 8–character journal name, which can be a specific or a generic name, for which this journal model can be used.

STREAMNAME
displays the MVS log stream name (LSN) associated with this journal model. The name can be a specific LSN, or a template using any 3 of the 4 symbols &USERID, &APPLID, &JNAME and &SYSID.

An LSN or template can be up to 26 characters in length.

TYPE
displays the log stream type. The possible types are:
MVS Records for journals defined with this journal model are written to an MVS log stream.
SMF Records for journals defined with this journal model are written to the MVS SMF data set.
DUMMY
Records for journals defined with this journal model are not written to any log stream.

For information on changing JMODEL definitions, see the *CICS Resource Definition Guide*.

CEMT INQUIRE JOURNALNAME

Show the status of the CICS system log and general logs.

Description

You can use the **INQUIRE JOURNALNAME** command to inquire on the status of the CICS system log and general logs. It replaces the INQUIRE JOURNALNUM command.

Input

Press the CLEAR key to clear the screen. You can start this transaction in two ways:

- Type `CEMT INQUIRE JOURNALNAME` (or suitable abbreviations for the keywords). The resulting display lists the current status.
- Type `CEMT INQUIRE JOURNALNAME` (or suitable abbreviations for the keywords), followed by the attributes that are necessary to limit the range of information that you require. For example, if you enter `cemt i jo sm`, the resulting display shows the details of only those logs that are of SMF log stream type.

(*value***)**
Specifies one or more journal names.

To inquire on journals defined with a numeric identifier in the range 1 - 99, specify journal name DFHJ*nn*, where *nn* is the journal number.

To inquire on the system log, specify DFHLOG.

To inquire on the log-of-logs, specify DFHLGLOG.

ALL
The default.

Sample screen

```
IN JOUR
STATUS:  RESULTS - OVERTYPE TO MODIFY
 Journaln(DFHJ02  ) Smf Ena
 Journaln(DFHLGLOG) Mvs Ena
    Str(CICSHT##.CICSHTA1.DFHLGLOG  )
 Journaln(DFHLOG  ) Mvs Ena
    Str(CICSHT##.CICSHTA1.DFHLOG    )
 Journaln(DFHSHUNT) Mvs Ena
    Str(CICSHT##.CICSHTA1.DFHSHUNT  )
```

Figure 78. CEMT INQUIRE JOURNALNAME screen

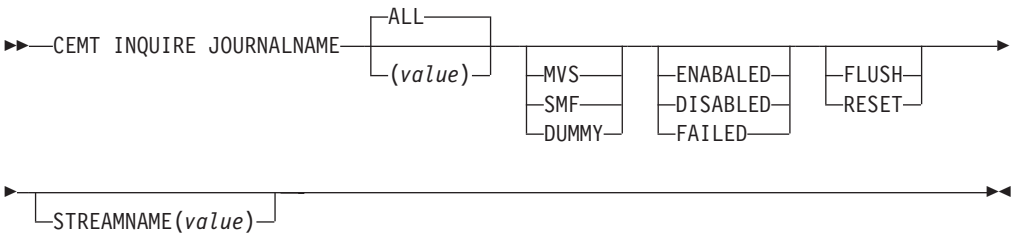
Note: Blank fields on the screen indicate that a value does not apply or is negative; that is, it begins with No.

If you place the cursor against a specific entry in the list, CICS displays an expanded format as shown in Figure 79.

```
IN JOUR
RESULT - OVERTYPE TO MODIFY
  Journalname(DFHLOG)
  Type(Smf)
  Status( Enabled )
  Action(             )
  Streamname(CICSHT##.CICSHTA1.DFHLOG)
```

Figure 79. The expanded display

CEMT INQUIRE JOURNALNAME



Displayed fields

ACTION (*value*)
Specifies the action you want CICS to take for the specified journal name. The values are:

FLUSH
The log buffers are written out to the log stream, but the journal is not closed.

You can use this option to ensure that all current records are written out to the log stream before the stream is processed using a batch utility.

RESET
The journal is disabled but can be re-enabled by the SET JOURNALNAME RESET command.

JOURNALNAME (*value*)
Indicates that this panel relates to a journal name inquiry and displays a 1- to 8-character journal name.

STATUS
Displays the status of the journal. The values are as follows:

ENABLED
The journal is installed and available for use.

DISABLED
The journal has been disabled by a CEMT, or EXEC CICS, SET JOURNALNAME(...) command. It cannot be used until it is re-enabled by the STATUS(ENABLED) or ACTION(RESET) options on a SET JOURNALNAME command.

FAILED
The journal has experienced a log stream failure. It cannot be used until it is re-enabled by the STATUS(ENABLED) or ACTION(RESET) options on a SET JOURNALNAME command, or following the next CICS restart. The log stream should be deleted from the MVS system logger inventory before being used again.

Note: You can reset STATUS with ENABLED or DISABLED.

STREAMNAME (*value*)
Displays the MVS log stream name associated with the journal. CICS displays 8 blanks (X'40') if the journal is defined with a type of DUMMY or SMF.

TYPE
Displays the type of log stream. The possible types are as follows:

MVS Records for journals defined with this journal model are written to an MVS log stream.

SMF Records for journals defined with this journal model are written to the MVS SMF data set.

DUMMY
Records for journals defined with this journal model are not written to any log stream.

CEMT INQUIRE JVM

Identify pooled JVMs in a CICS region and report their status.

 In the CICS Explorer, the JVM status operations view provides a functional equivalent to this command.

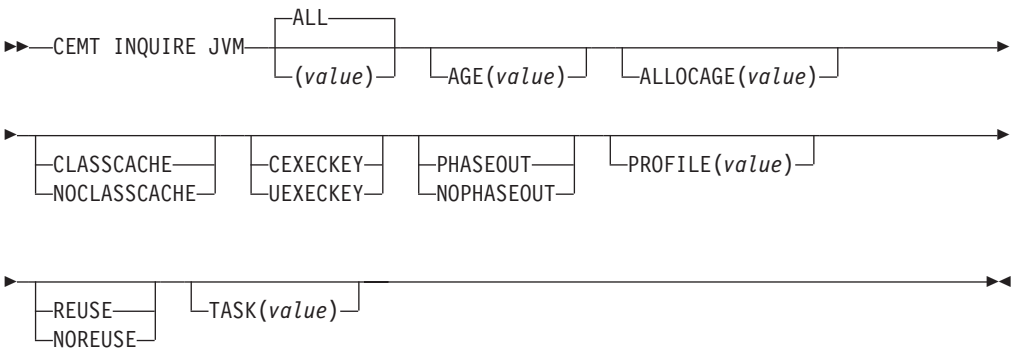
Input

Enter `CEMT INQUIRE JVM` to display all the pooled JVMs in the CICS region. You cannot change the displayed values. You can also enter `CEMT INQUIRE JVM` with a

filter to display all the JVMs in the CICS region with a particular status. For example, you can enter `CEMT INQUIRE JVM UEXECKEY` to display all the pooled JVMs that run in user key, or `CEMT INQUIRE JVM PROFILE(DFHJVMPR)` to display all the pooled JVMs that were created with the JVM profile DFHJVMPR.

You can use this command only for pooled JVMs. If you want to find out about a JVM server, use the CEMT INQUIRE JVMSERVER command.

CEMT INQUIRE JVM



Displayed fields

(*value***)**
The JVM number, a unique value that identifies the JVM. The JVM number used in CICS is the same number that is used to identify the JVM in the z/OS UNIX environment, where it is known as the process ID (PID) for the JVM.

AGE(*value***)**
Displays the number of seconds since the JVM was initialized.

ALLOCAGE(*value***)**
Displays the number of seconds for which the JVM has been allocated to its task, or zero if the JVM is not currently allocated to a task.

CLASSCACHEST
Displays whether the JVM uses the shared class cache:

CLASSCACHE
The JVM profile for this JVM specified the use of the shared class cache.

NOCLASSCACHE
The JVM profile for this JVM did not specify the use of the shared class cache.

EXECKEY
Displays the execution key of the JVM:

CEXECKEY
The JVM runs in CICS key.

UEXECKEY
The JVM runs in user key.

PHASINGOUTST
Displays whether the JVM is being phased out as a result of a **PHASEOUT**, **PURGE**, or **FORCEPURGE** command:

PHASEOUT
The JVM is being phased out.

NOPHASEOUT
The JVM is not being phased out. It is currently available for allocation or is available for allocation when the current allocation is ended.

PROFILE(*value***)**
Displays the 8-character name of the JVM profile used to initialize this JVM.

REUSEST
Displays whether this JVM can be reused:

REUSE
The JVM profile for this JVM specified that it is continuous.

NOREUSE
The JVM profile for this JVM specified that it is single-use.

TASK(*value***)**
Displays the task to which the JVM is allocated or zero if the JVM is not currently allocated to a task.

CEMT INQUIRE JVMPOOL

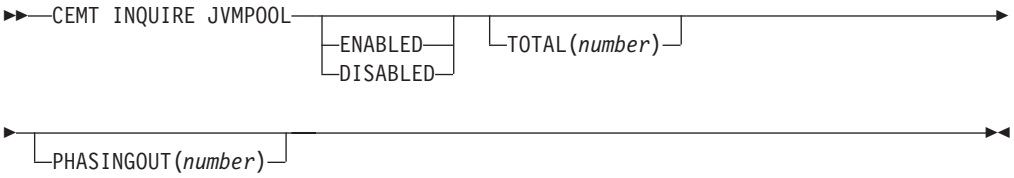
Retrieve information about the JVM pool in the CICS region.

In the CICS Explorer, the JVM pools operations view provides a functional equivalent to this command.

Input

Enter CEMT INQUIRE JVMPOOL to obtain information about the pool of JVMs in the CICS region. Only one pool of JVMs exists in a CICS region and therefore no name or identifier is required on this command. This command does not return information about JVM servers. For information about JVM servers, see "CEMT INQUIRE JVMSERVER" on page 251.

CEMT INQUIRE JVMPOOL



Displayed fields

PHASINGOUT(*number***)**
Displays the number of JVMs that are marked for removal from the JVM pool. These JVMs are still allocated to a task that is currently running, or has run, a Java program in the JVM.

STATUS(*value***)**
Displays the overall status of the JVM pool:
ENABLED
The pool is enabled for use and Java programs can run using JVMs from the pool. This value is the normal status.

DISABLED
The pool is disabled and new requests cannot be serviced from the pool. Programs can still be running if they were started before the JVM pool became disabled.

TOTAL (*number***)**
Displays the number of JVMs that have been initialized and are available for use or allocated to tasks. This total includes JVMs that are in the process of being terminated and removed from the region and included in the PHASINGOUT count.

CEMT INQUIRE JVMSERVER

Retrieve information about the status of a JVM server runtime environment.

In the CICS Explorer, the JVM servers operations view provides a functional equivalent to this command.

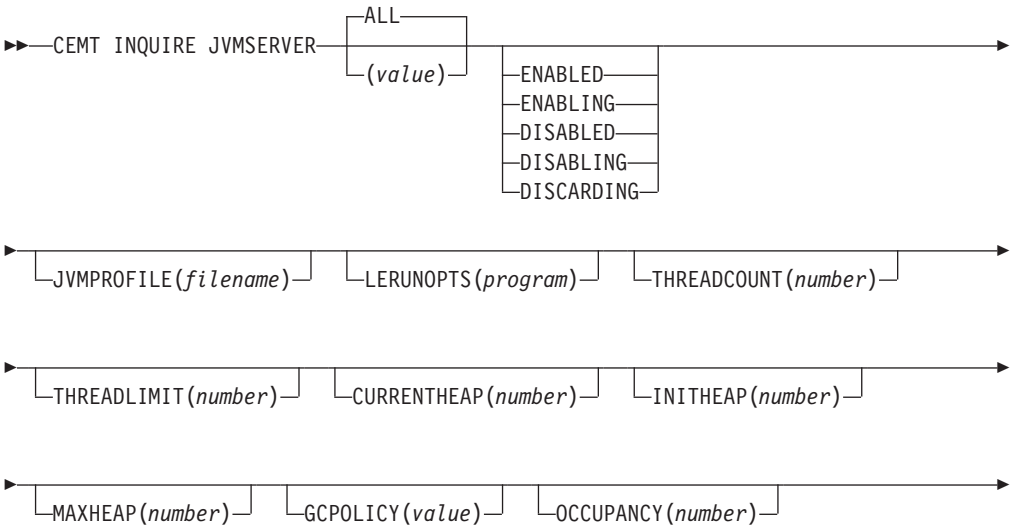
The resource signature

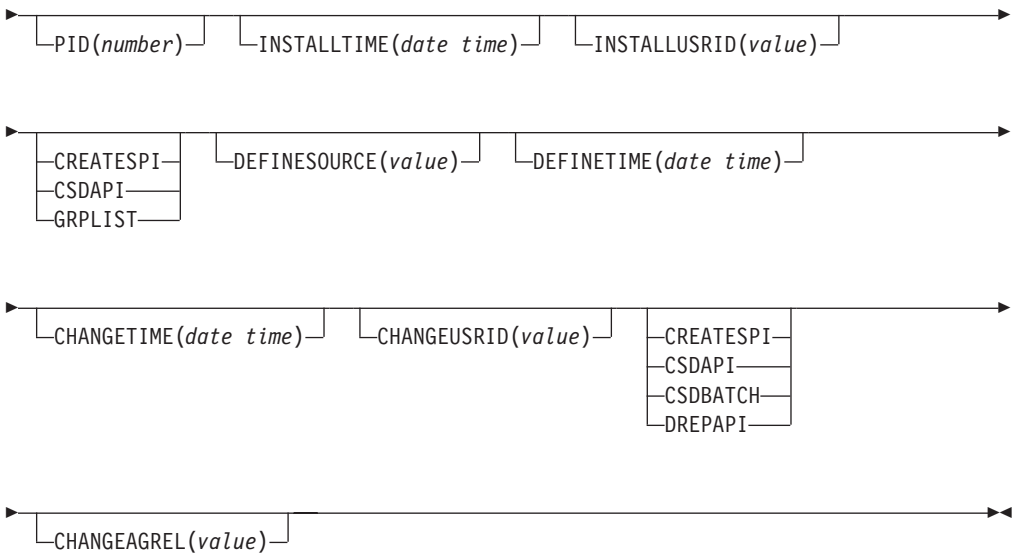
You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key to clear the screen and then enter `CEMT INQUIRE JVMSERVER` to obtain information about a JVM server.

CEMT INQUIRE JVMSERVER





Displayed fields

CHANGEAGENT(*value*)
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

CHANGEAGREL(*value*)
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time*)
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value*)
Displays the 8-character user ID that ran the change agent.

CURRENTHEAP(*number*)
Displays the current heap size in bytes that is allocated to the JVM server.

DEFINESOURCE(*value*)
Displays the source of the resource definition. The DEFINESOURCE value

depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

GCPOLICY(*value***)**
Displays the garbage collection policy that is being used by the JVM server.

INITHEAP(*number***)**
Displays the initial heap size in bytes that is allocated to the JVM server.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

JVMPROFILE(*filename***)**
Displays the file name of the JVM profile that defines the properties of the JVM server.

LERUNOPTS(*program***)**
Displays the name of the program that defines the runtime options for the Language Environment enclave.

MAXHEAP(*number***)**
Displays the maximum heap size in bytes that is allowed in the JVM server.

OCCUPANCY(*number***)**
Displays the heap size in bytes after the last garbage collection ran in the JVM server.

PID(*number***)**
Displays the process ID (PID) of the JVM.

STATUS(*value***)**
Displays the overall status of the JVM server runtime environment. The possible values are as follows:
ENABLED
The JVM server is enabled and can perform processing.

ENABLING
The JVM server is starting.
DISABLED
The JVM server is disabled and new requests cannot be processed in the runtime environment.
DISABLING
The JVM server is stopping. The JVM server can complete any work that began before it is disabled.
DISCARDING
The JVMSERVER resource is being discarded.

THREADCOUNT(*number***)**
Displays the number of threads that are currently running inside the JVM server.

THREADLIMIT(*number***)**
Displays the maximum number of threads that can be used by the JVM server. Each thread runs under a T8 TCB.

CEMT INQUIRE LIBRARY

Retrieve information about LIBRARY resources.

In the CICS Explorer, the Libraries operations view provides a functional equivalent to this command.

Description

The **CEMT INQUIRE LIBRARY** command returns information about the LIBRARY resources that are defined to your system.

You can reset the options either by typing the SET command or by overtyping at the appropriate place on the INQUIRE screen.

The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

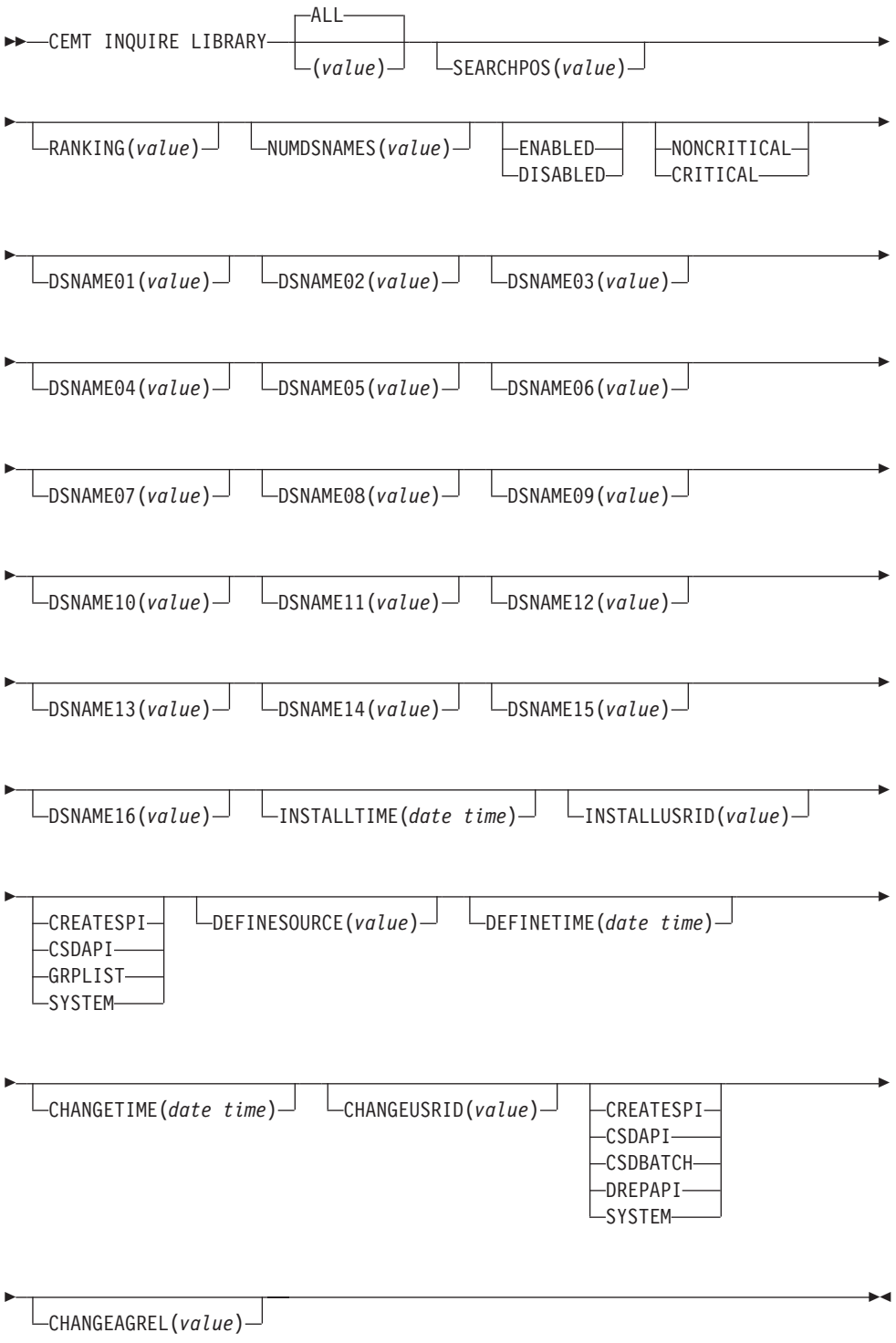
Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT INQUIRE LIBRARY` (or suitable abbreviations for the keywords). The resulting display lists the current status. LIBRARY resources are listed in ranking order, and by time of install in libraries of equal ranking.
- Type `CEMT INQUIRE LIBRARY` (or suitable abbreviations for the keywords), followed by the attributes that are necessary to limit the range of information required.

You can tab to the highlighted or blank fields and overtype them with the values required.

(*data-value***)**
Specifies the name of the LIBRARY resource. If you omit the LIBRARY name, the ALL option is assumed by default.

CEMT INQUIRE LIBRARY



Displayed fields

CHANGEAGENT(*value*)
Displays a value that identifies the agent that made the last change to the

resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

SYSTEM
The resource definition was last changed by the CICS or CICSPlex system.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

CRITICALST(*value***)**
Indicates whether the library is critical to the startup of CICS:

CRITICAL
The LIBRARY is critical to CICS startup. If the LIBRARY cannot be successfully installed during CICS startup, a GO or CANCEL message is issued and the operator must decide whether to override the criticality and allow CICS to start. If the operator selects GO, the library is installed in a disabled status unless installation is not possible at all; for example, because of a short-on-storage condition. If the reply is to continue with the startup, the library is not recatalogued as NONCRITICAL, so the critical status must be explicitly set to NONCRITICAL if you decide that the LIBRARY will not be regarded as CRITICAL in future.

NONCRITICAL
The LIBRARY is not critical to CICS startup. If the LIBRARY cannot be successfully installed during CICS startup, the LIBRARY is left in an installed but disabled state and a warning message is issued, but CICS startup continues.

LIBRARY(*data-value***)**
Specifies the 8-character name of the LIBRARY about which you are inquiring. If you omit the program name, the ALL option is assumed by default.

DEFINESOURCE(*value***)**
Displays the source of the resource definition. The DEFINESOURCE value

depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

DSNAME01-16(*value***)**
Return the names of data sets in the LIBRARY concatenation. If this library is dynamically-defined, these are the data sets specified on the LIBRARY definition, all but one of which can be blank. If this DFHRPL is statically-defined, these are the first 16 data sets in the DFHRPL concatenation, or as many data sets as are specified up to 16, with the remaining DSNAMExx fields being blank.

If there are more than 16 data sets in the DFHRPL concatenation, you can use either a CICSPlex® SM LIBDSN view to list all of the data sets or the DSNAMELIST option on the **EXEC CICS INQUIRE LIBRARY** command.

ENABLESTATUS(*value***)**
Indicates whether the LIBRARY is included in the overall LIBRARY search order:

DISABLED
The LIBRARY is disabled, and is not currently included in the LIBRARY search order. The data sets in this LIBRARY concatenation are not searched for program artifacts.

DISABLING
A request to disable the LIBRARY has been received, but is still being processed.

ENABLED
The LIBRARY is enabled, and is currently included in the LIBRARY search order. The data sets in this LIBRARY concatenation will be searched for program artifacts to load.

ENABLING
A request to enable the LIBRARY has been received, but is still being processed.

DISCARDING
A request to discard the LIBRARY from the CICS system has been received, but is still being processed.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

SYSTEM
The resource was installed by the CICS or CICSPlex SM system.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

NUMDSNAMES(*value***)**
Specifies the number of data sets in the LIBRARY concatenation. For a dynamically defined LIBRARY, this value is the number of non blank DSNAMExx values, and cannot be a value larger than 16. For the statically defined DFHRPL, this value is the number of data sets in the concatenation, and can be a value larger than 16.

RANKING(*value***)**
Specifies where this LIBRARY appears in the overall library search order relative to other library concatenations. A lower number indicates that this LIBRARY is searched for programs to load before other library resources with higher ranking numbers. However libraries of equal RANKING will appear in the search order in the order in which they were installed or created in the local CICS system, with a library that was installed earlier appearing before one that was installed later.

SEARCHPOS(*data-area***)**
Returns the current absolute position of this LIBRARY in the overall LIBRARY search order. The first LIBRARY in the search order has a SEARCHPOS of 1, the next LIBRARY has a SEARCHPOS of 2, and so on. The SEARCHPOS is not the same as the RANKING, although its value is determined by the relative ranking values of the various LIBRARY resources in the system.

CEMT INQUIRE LINE

Retrieve information about lines connected to terminals. This request is valid only for non-z/OS Communications Server terminals.

Input

Press the Clear key to clear the screen. There are two ways to start this transaction:

- Type `CEMT INQUIRE LINE(value)` (or suitable abbreviations for the keywords). The current status of the named line is displayed.
- Type `CEMT INQUIRE LINE(termid)` (or suitable abbreviations for the keywords), followed by as many attributes as appropriate to limit the range of information that you require.

You can specify a list of identifiers (see “List of resource identifiers” on page 116), but you cannot use the asterisk (*) or plus (+) symbols to specify a family of identifiers. You can tab to the highlighted or blank fields and overtype them with the required values.

Sample screen

```
INQ LINE(TTT1)
STATUS:  RESULTS
 Lin(TTT1)         Neg( 00080 ) Ins
```

Figure 80. CEMT INQUIRE LINE screen

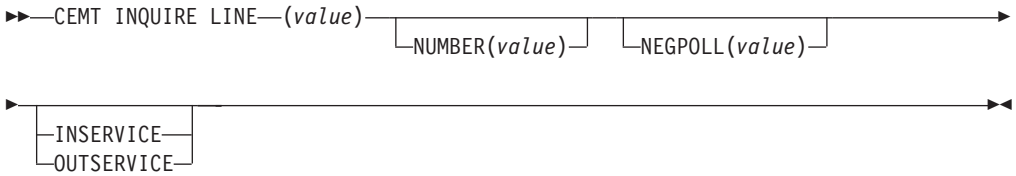
Note: The screen has blank fields where a value does not apply or is negative; that is, the value begins with "No". To modify such fields, tab to locate each one (they appear in the same sequence as in the expanded format), and overtype with input valid for that field. It can be easier to use the expanded format to set such a value (setting a value negative is easier because you overtype a displayed field with its negative, which results in a blank field).

If you place the cursor against a specific entry in the list, CICS displays an expanded format as shown in Figure 81.

```
INQ LINE(TTT1)
RESULT
  Line(TTT1)
  Number()
  Negpoll( 00080 )
  Servstatus( Inservice )
  Purgetype(            )
```

Figure 81. The expanded display

CEMT INQUIRE LINE



Displayed fields

LINE (*value*)
Indicates that this panel relates to a LINE inquiry and displays the identifier of a terminal on the line. This request is valid only for non-z/OS Communications Server terminals.

NUMBER (*value*)
Displays the relative line number of a pooled line. For information about pooled lines, see the *CICS Resource Definition Guide*.

SERVSTATUS
The possible values are as follows:

INSERVICE
The line is available for use. Changing the status of the line does not affect the equivalent status of the terminals on it.

OUTSERVICE
The line is not available for use. Changing the status of the line does not affect the equivalent status of the terminals on it.

CEMT INQUIRE MODENAME

Retrieve information about connection sessions.

Description

INQUIRE MODENAME returns information about the named group of sessions that has been defined for a connection to a remote system, or to another CICS region.

Input

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT INQUIRE MODENAME` (or suitable abbreviations for the keywords). The resulting display lists the current status.
- Type `CEMT INQUIRE MODENAME` (or suitable abbreviations for the keywords), followed by the attributes that are necessary to limit the range of information that you require. For example, if you enter `cemt i mod acq`, the resulting display shows the details of all mode names that are acquired.

Mode names do not need to be unique. You cannot specify a list of names, but you can use the asterisk (*) and plus (+) symbols to specify a family of mode names. A mode name consisting of eight blanks is valid. To specify this name, enclose the string of blanks in single quotation marks, for example:

```
MODENAME('   ')
```

You can then tab to the highlighted or blank fields and overtype them with the required values.

(*value***)**
Specifies one or more names (1 - 8 characters) of a group of sessions.

ALL
The default.

Sample screen

```
I MODE
STATUS:  RESULTS - OVERTYPE TO MODIFY
 Mod(SNASVCMG) Con(C   ) Max(002) Ava( 000 ) Act(000)
 Mod(MODE1   ) Con(C   ) Max(001) Ava( 000 ) Act(000)
```

Figure 82. CEMT INQUIRE MODENAME screen

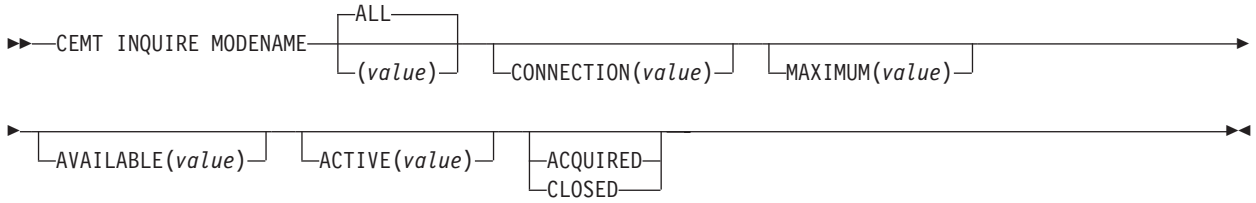
Note: When a value does not apply, or is negative (the value begins with No), the fields on the screen are blank. To modify these fields, locate them by tabbing (they appear in the same sequence as in the expanded format), and overtype with input valid for that field. You might find it more convenient to use the expanded format when setting one of these values.

If you place the cursor against a specific entry in the list, CICS displays an expanded format, as shown in Figure 83 on page 262.

```
I MODE
RESULT - OVERTYPE TO MODIFY
  Modename(MODE1)
  Connection( C    )
  Maximum(001)
  Available( 000 )
    Active(000)
    Acqstatus(             )
```

Figure 83. The expanded display

CEMT INQUIRE MODENAME



Displayed fields

ACQSTATUS (*input only field*)
Specifies the acquisition status of sessions. The values are as follows:

ACQUIRED
CICS acquires (or binds) all the negotiated contention-winner sessions in the group.

CLOSED
CICS releases all sessions in the group. The sessions are quiesced and unbound. No further sessions can be acquired by another CICS system. No further activity is possible on the sessions in the group until the group is reacquired by the same CICS system in which it has been closed.

ACTIVE (*value*)
Displays the number of sessions in the group (or mode name) that are currently in use (bound).

AVAILABLE (*value*)
Displays the currently negotiated number of sessions in the group that can be allocated for use at one time.

You can reset this value by overtyping it with a different value.

CONNECTION (*value*)
Displays the 4-character identifier of the remote system with which this group of IRC or ISC sessions is connected.

MAXIMUM (*value*)
Displays the maximum number of sessions that are supported at one time in this group of sessions, in the range 0 - 999.

MODENAME (*value*)
Indicates that this panel relates to a MODENAME inquiry and displays one or more names (1 - 8 characters) of a group of sessions.

CEMT INQUIRE MONITOR

Retrieve information about the status of CICS monitoring.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

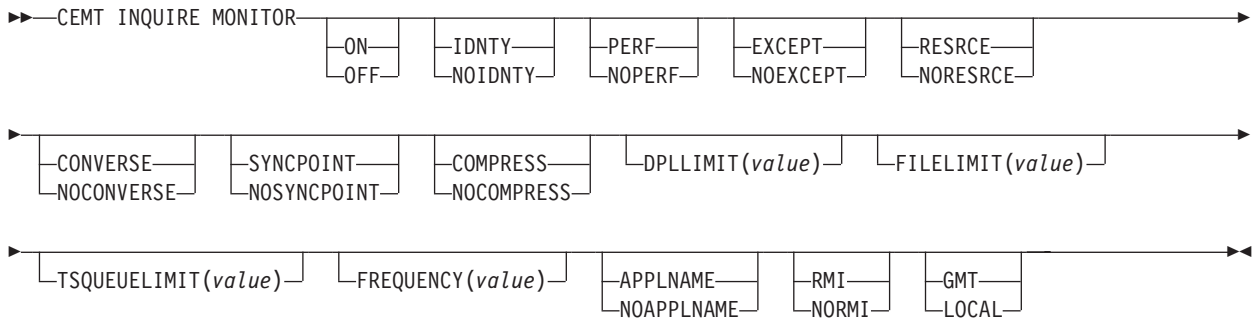
You can use the INQUIRE MONITOR command to inquire whether monitoring is switched on or off, which classes of monitoring data are collected, and the current settings of the monitoring options.

Input

Press the Clear key to clear the screen, and type `CEMT INQUIRE MONITOR` (or suitable abbreviations for the keywords). The resulting display lists the current status. You can tab to the highlighted or blank fields and overtype them with the required values.

Displayed fields

CEMT INQUIRE MONITOR



APPLNAMEST(*value*)
Displays whether CICS application naming support is enabled. The values are as follows:
APPLNAME
CICS application naming support is enabled.
NOAPPLNAME
CICS application naming support is not enabled.

COMPRESSST(*value*)
Displays whether data compression is performed for monitoring records. The values are as follows:
COMPRESS
Data compression is performed. The default is for monitoring records to be compressed.
NOCOMPRESS
Data compression is not performed.

You can reset this value by overtyping it with a different value.

CONVERSEST(*value*)
Displays whether conversational tasks have separate performance class records produced for each pair of terminal control I/O requests (Converse or Send/Receive pair). The values are as follows:
CONVERSE
Separate performance class records are produced.
NOCONVERSE
Separate performance class records are not produced.

You can reset this value by overtyping it with a different value.

DPLLIMIT (*value*)
Displays the maximum number of distributed program link requests for which transaction resource class monitoring data is being collected.

EXCEPTCLASS(*value*)
Displays whether the exception class of monitoring data is being collected. The values are as follows:
EXCEPT
Exception-class monitoring data is being collected.
NOEXCEPT
Exception-class monitoring data is not being collected.

You can reset this value by overtyping it with a different value.

FILELIMIT (*value*)
Displays the maximum number of files for which transaction resource class monitoring data is being collected.

FREQUENCY (*value*)
Displays the interval for which monitoring automatically produces a transaction performance class record for any long-running transaction that has been in the system for a greater time than the interval. FREQUENCY values can be either 000000 (indicating that FREQUENCY monitoring is inactive) or in the range 000100 – 240000.

You can reset this value by overtyping it with a different value.

MONITOR(*value*)
Indicates that this panel relates to a MONITOR inquiry.

IDNTYCLASS(*value*)
Displays whether the identity class of monitoring data is being collected. The values are as follows:

IDNTY
Identity class monitoring data is being collected.

NOIDNTY
Identity class monitoring data is not being collected.

PERFCLASS(*value*)
Displays whether the performance class of monitoring data is being collected. The values are as follows:
PERF Performance class monitoring data is being collected.
NOPERF
Performance class monitoring data is not being collected.

You can reset this value by overtyping it with a different value.

RESRCECLASS(*value*)
Displays whether the transaction resource class of monitoring data is being collected. The values are as follows:

RESRCE
Transaction resource class monitoring data is being collected.
NORESRCE
Transaction resource class monitoring data is not being collected.

You can reset this value by overtyping it with a different value.

RMIST(*value***)**
Displays whether additional performance class monitoring data is to be collected for the resource managers used by your transactions. The values are as follows:
RMI Additional performance class monitoring data is being collected for the resource managers used by your transactions.
NORMI
Performance class monitoring data is not being collected for the resource managers used by your transactions.

STATUS(*value***)**
Displays whether monitoring is active. The values are as follows:
ON Monitoring data is accumulated continuously and is collected (written out to an SMF data set) for each of the classes of monitoring data that is active.
OFF No monitoring data is accumulated or collected, regardless of the settings of the monitoring data classes.

You can reset this value by overtyping it with a different value.

SYNCPOINTST(*value***)**
Displays whether a transaction performance class record is produced when a transaction takes an explicit or implicit sync point (unit-of-work), except when the sync point is part of task termination or a sync point rollback. The values are as follows:
SYNCPOINT
A transaction performance class record will be produced.
NOSYNCPOINT
A transaction performance class record will not be produced.

You can reset this value by overtyping it with a different value.

TIME(*value***)**
Displays whether the time stamp fields in the performance class monitoring data are returned in Greenwich mean time (GMT) or local time to an application using the **COLLECT STATISTICS** command. The values are as follows:
GMT The time-stamp fields in the performance class data are GMT.
LOCAL
The time-stamp fields in the performance class data are local time.

TSQUEUELIMIT(*value***)**
Displays the maximum number of temporary storage queues for which transaction resource class monitoring data is being collected.

CEMT INQUIRE MQCONN

Retrieve information about the connection between CICS and WebSphere MQ.

In the CICS Explorer, the Websphere MQ Connections operations view provides a functional equivalent to this command.

Description

The **INQUIRE MQCONN** command returns information about the attributes of the currently installed MQCONN resource definition, which defines the connection to WebSphere MQ, and about the status of the connection.

Because only one MQCONN resource definition can be installed at a time, the name of the MQCONN resource definition is not required on input.

This command does not inquire on the INITQNAME attribute of the MQCONN resource definition, which specifies the name of the initiation queue. Use the INQUIRE MQINI command to inquire on the initiation queue name.

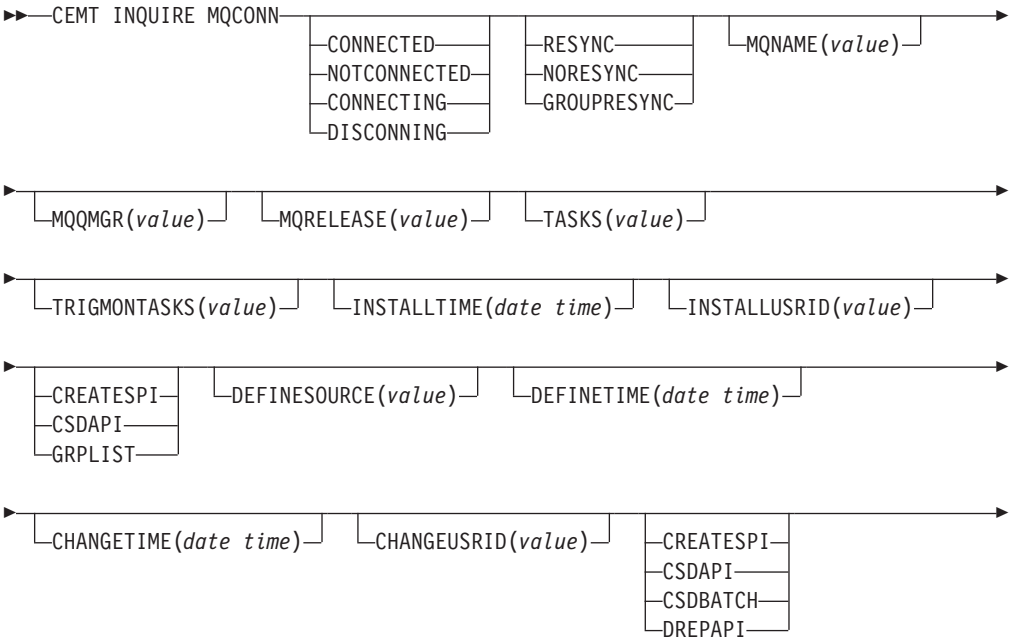
The resource signature

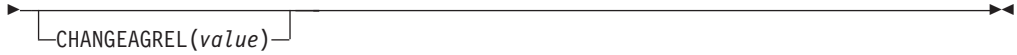
You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key to clear the screen, and type `CEMT INQUIRE MQCONN` (or suitable abbreviations for the keywords). The resulting display lists the status. You can tab to the highlighted or blank fields and overtype them with the required values.

CEMT INQUIRE MQCONN





Displayed fields

CHANGEAGENT(*value***)**
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

CONNECTST(*value***)**
Displays the status of the CICS-MQ connection. The possible values are as follows:

CONNECTED
CICS is connected to WebSphere MQ.

NOTCONNECTED
CICS is not connected to WebSphere MQ.

CONNECTING
CICS is currently attempting to connect to WebSphere MQ.

DISCONNING
CICS is currently disconnecting from WebSphere MQ.

DEFINESOURCE(*value***)**
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the

date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

MQNAME(*value***)**
Displays the 1- to 4-character name of the WebSphere MQ queue manager or queue-sharing group that you specified in the MQCONN resource definition (or by using a SET MQCONN command) for the CICS region.

MQQMGR(*value***)**
Displays the name of the WebSphere MQ queue manager to which CICS is connected or to which CICS is waiting to connect.

- If CICS is connected to WebSphere MQ, this field shows the name of the queue manager to which CICS is connected. If you specified a queue-sharing group in the MQCONN resource definition for the CICS region, the queue manager shown here is the one that was chosen from the group.
- If CICS is not connected to WebSphere MQ, this field usually contains blanks. However, if you specified a queue-sharing group in the MQCONN resource definition for the CICS region, and CICS is waiting to reconnect to a specific queue manager in the queue-sharing group because it is holding outstanding units of work for that queue manager, the name of the specific queue manager is shown, and the status of the connection is shown as CONNECTING. For this situation to arise, the RESYNCMEMBER attribute in the MQCONN resource definition must specify resynchronization.

MQRELEASE(*value***)**
If CICS is connected to WebSphere MQ, this field displays the release number of WebSphere MQ; for example, 0600. When CICS is not connected to WebSphere MQ, the field is blank.

RESYNCMEMBER(*value***)**
This option applies only if you have specified a queue-sharing group for the CICS-MQ connection. RESYNCHMEMBER displays the strategy that CICS adopts if outstanding units of work are being held for the last queue manager to which CICS was connected from the queue-sharing group. Units of work that are shunted indoubt are not included in this process, because CICS itself

cannot resolve those units of work at this time. Resynchronization for those UOWs occurs when CICS has resynchronized with its remote coordinator. The possible values are as follows:

RESYNC
CICS connects to the same queue manager.

NORESYNC
CICS makes one attempt to connect to the same queue manager. If that attempt fails, CICS connects to any member of the queue-sharing group and issues a warning about the outstanding units of work.

GROUPRESYNC
CICS connects to any member of the queue-sharing group. The queue manager is chosen by WebSphere MQ and it asks CICS to resolve indoubt units of work on behalf of all eligible queue managers in the queue-sharing group. This function is called *group unit of recovery*.

NOTAPPLIC
A queue-sharing group is not specified for the CICS-MQ connection.

TASKS(*value***)**
Displays the current number of tasks that are using the CICS-MQ connection, including trigger monitor tasks.

TRIGMONTASKS(*value***)**
Displays the current number of trigger monitor tasks that are using the CICS-MQ connection.

CEMT INQUIRE MQINI

Retrieve the name of the default initiation queue used for the connection between CICS and WebSphere MQ.

In the CICS Explorer, the Websphere MQ Initiation Queues operations view provides a functional equivalent to this command.

Description

The **INQUIRE MQINI** command inquires on the name of the default initiation queue used for the connection between CICS and WebSphere MQ.

The MQINI resource represents the initiation queue. MQINI is an implicit resource that exists when you install an MQCONN resource definition in the CICS region with the INITQNAME attribute specified. Only one MQCONN resource definition can be installed at a time. The name of the MQINI resource is DFHMQINI.

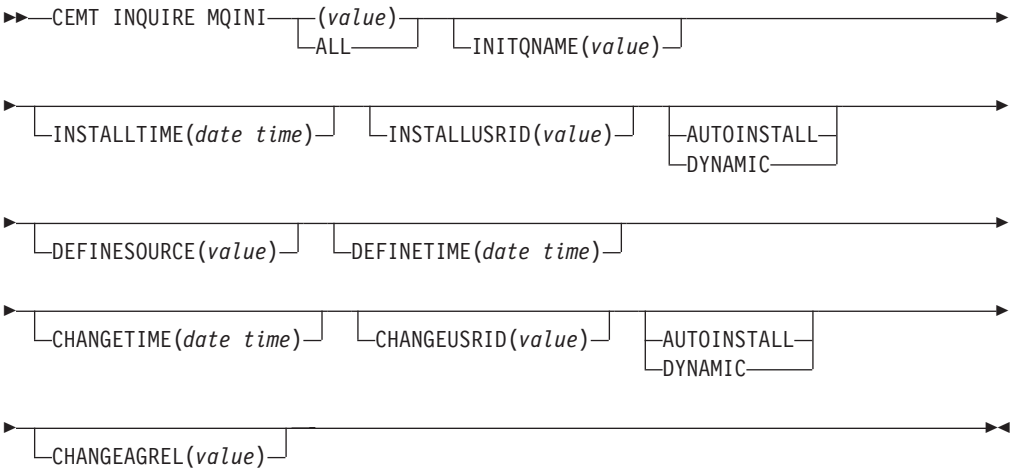
The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key to clear the screen, and type CEMT INQUIRE MQINI (or suitable abbreviations for the keywords). The resulting display lists the current status. You can tab to the highlighted or blank fields and overtype them with the required values. To view the display in an expanded format, place the cursor against the resource name and press Enter.

CEMT INQUIRE MQINI



Displayed fields

CHANGEAGENT(*value*)
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

AUTOINSTALL
The resource was autoinstalled as a result of specifying an initiation queue name on a CKQC START command, and the previously installed MQCONN definition did not specify a value for INITQNAME.

DYNAMIC
The resource was defined as a result of an MQCONN resource definition with INITQNAME specified.

CHANGEAGREL(*value*)
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time*)
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value*)
Displays the 8-character user ID that ran the change agent.

DEFINESOURCE(*value*)
Displays the source of the resource definition. The DEFINESOURCE value

depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INITQNAME(*value***)**
Displays the 1- 48-character name of the default initiation queue that is represented by the MQINI resource.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. The possible values are as follows:

AUTOINSTALL
The resource was autoinstalled as a result of specifying an initiation queue name on a CKQC START command, and the previously installed MQCONN definition did not specify a value for INITQNAME.

DYNAMIC
The resource was installed as a result of the installation of an MQCONN with INITQNAME specified.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

CEMT INQUIRE NETNAME

Retrieve information about a network.

In the CICS Explorer, the Terminals operations view provides a functional equivalent to this command.

Description

If you are familiar with network names, you might prefer to use this command as an alternative to the **CEMT INQUIRE TERMINAL** command.

Input

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT INQUIRE NETNAME` (or suitable abbreviations for the keywords). The resulting display lists the status.
- Type `CEMT INQUIRE NETNAME` (or suitable abbreviations for the keywords), followed by the attributes that are necessary to limit the range of information that you require. For example, if you enter `cemt i n ins ati`, the result shows details of network names that are in service and available for use.

You can tab to the highlighted or blank fields and overtype them with the required values.

(*value***)**
Specifies an 8-character network name.

ALL
The default.

Sample screen

```
IN NETNAME
STATUS:  RESULTS - OVERTYPE TO MODIFY
 Net(IGKS205 ) Tra(CEMT) Pri( 000 ) Pag Ins Ati Tti
         Ter(TC03) Acq     Tas(0000198)           Nqn(GBIBMIYA.IYCQTC03)
```

Figure 84. CEMT INQUIRE NETNAME screen

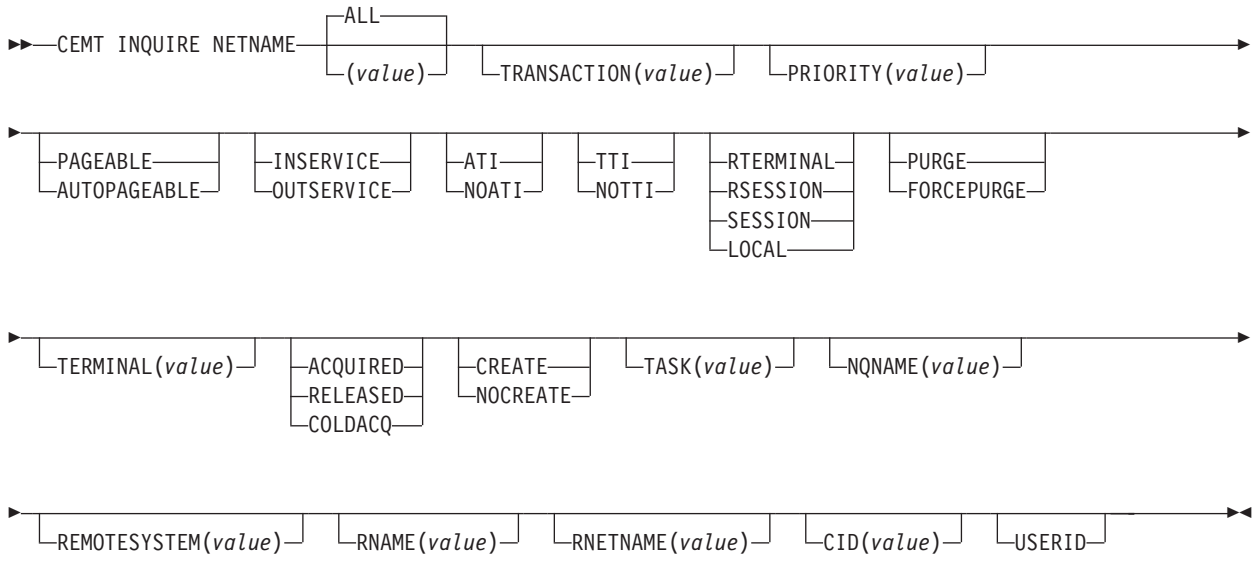
Note: When a value does not apply, or is negative (the value begins with No), the fields on the screen are blank. To modify these fields, locate them by tabbing (they appear in the same sequence as in the expanded format), and overtype with input valid for that field. You might find it more convenient to use the expanded format when setting one of these values.

If you place the cursor against a specific entry in the list and press ENTER, CICS shows an expanded format, as shown in Figure 85.

```
IN NET
RESULT - OVERTYPE TO MODIFY
  Netname(IYCQTC03)
  Transaction(CEMT)
  Priority( 000 )
  Pagestatus( Pageable )
  Servstatus( Inservice )
  Atistatus( Ati )
  Ttistatus( Tti )
  Nature(Local)
  Purgetype(              )
  Terminal(944D)
  Termstatus( Acquired )
  Createsess( Nocreate )
  Task(00033)
  Remotesystem()
  Nqname(GBIBMIYA.IYCQTC03)
  Rname()
  Rnetname()
  Cid()
  Userid()
```

Figure 85. The expanded display of an individual entry

CEMT INQUIRE NETNAME



Displayed fields

AIDCOUNT (*data area***)**
Returns a fullword binary field that contains the number of automatic initiate descriptor (AIDs) queued for the specified terminal. If there are no AIDs, a value of 0 is returned. The count represents the number of AIDs on the CICS system that issued the command.

ATISTATUS
Displays whether the terminal is available for use by transactions that are automatically initiated from within CICS or, if the terminal is an ISC session, by transactions that are using this session as an alternate facility to communicate with another system. The values are as follows:
ATI The terminal is available for use.
NOATI
The terminal is not available for use.

Note:

1. You can reset this value by overtyping it with a different value.
2. A terminal cannot be defined with both NOATI and NOTTI.

CID (*value***)**
Displays an 8-character correlation identifier that is set only for sessions, and only after a session is acquired. (The value relates to the previously acquired session if there was one, if the session is not acquired). CID is set as follows:

- For LU6.2 sessions, it is an 8-character hexadecimal token that is common to the two end sessions that are connected together.
- For MRO sessions, it is set to the termid of the session on the system to which this session is connected.
- For LU6.1 sessions to IMS, it is set to the value of NETNAMEQ as defined or negotiated.

- For other LU6.1 sessions, it is set to the value supplied by the other end of the 6.1 session.

Using CID, you can relate the two parts of an LU6.2, MRO, or LU61 conversation together:

- For LU6.2, you can inquire on the same CID.
- For MRO, you can locate the session with the terminal name.
- For LU6.1 to IMS, you can inquire on the netname.
- For other LU6.1, you can use the name to inquire on the connected system.

CREATESESS (*z/OS Communications Server only***)**
Displays whether the terminal can be acquired automatically by ATI transactions. This cannot be specified for IRC sessions. The values are as follows:

CREATE
If the terminal is not in session, CICS acquires it if the terminal is required for an ATI request.

NOCREATE
If the terminal is not in session, CICS does not acquire it to satisfy an ATI request. A session must be started by, for example, a logon request or a CEMT SET TERMINAL ACQUIRED command before the ATI request can be satisfied.

If NOCREATE is set for an LU6.1 ISC session and there are no allocatable sessions left, the connection is placed OUTSERVICE.

You can reset this value by overtyping it with a different value.

NATURE
Displays the nature of the terminal; that is, whether it is a physical device or a session, and whether it is local or remote. The values are as follows:

RTERMINAL
The terminal is owned by another CICS region.

RSESSION
The terminal is a session of a connection that is owned by another CICS region.

SESSION
The terminal is an ISC or IRC session.

LOCAL
The terminal is directly attached to this system and is a physical device or a function of a cluster controller.

NETNAME (*value***)**
Indicates that this panel relates to a NETNAME inquiry and displays an 8-character network name.

- For a physical terminal, the NETNAME is the name by which this terminal is known to z/OS Communications Server.
- For ISC sessions, the NETNAME is the name by which the session (or group of sessions) is known to z/OS Communications Server.
- For IRC sessions, the NETNAME is the name used by the connected region to logon to the interregion communication program (DFHIRP).
- For EXCI sessions, the NETNAME is DFHGEN for generic sessions and, for specific sessions, it is the value of the user_name parameter used in the EXCI INITIALISE_USER call.

When an inquiry is made on DFHGEN, the name is an 8-character field with a leading and trailing blank (' DFHGEN '). The name must be enclosed in quotes and include the blanks, so you enter the command on the screen as follows:

```
CEMT INQ NETNAME(' DFHGEN ')
```

- For remote devices, the NETNAME is the name by which the device is known to the z/OS Communications Server in the terminal-owning region.

Note: If the NETNAME is a z/OS Communications Server LU alias, it is different from the netname component of the NQNAME, which always contains the real netname.

NQNAME
Displays the z/OS Communications Server 17-character network-qualified name.

Remote terminals do not have an NQNAME value.

If the status (Termstatus) is RELEASED, NQNAME displays the netname.

PAGESTATUS
Displays whether pages after the first in a series are written to the terminal on request from the operator or automatically. The values are as follows:

PAGEABLE
Pages are written on request.

AUTOPAGEABLE
Pages are written automatically.

You can reset this value by overtyping it with a different value.

PRIORITY (*value*)
Displays a 3-character string that indicates the priority of this terminal relative to other terminals. The priority of a task is the sum of the transaction priority, the terminal priority, and the operator priority. Priority has no meaning for terminals that are ISC sessions being used as alternative facilities. The value is in the range 0 - 255, where 255 is the highest priority.

You can reset this value by overtyping it with a different value.

PURGETYPE (*input only field*)
Specifies whether the transactions running with the named terminal are to be purged. The values are as follows:

PURGE
Transactions are to be terminated only if system and data integrity can be maintained. A transaction is to be purged if its definition specifies SPURGE=NO.

FORCEPURGE
Transactions are to be purged immediately. This can lead to unpredictable results and is for use in exceptional circumstances.

REMOTESYSTEM (*value*)
Displays the 4-character name of a connection, if the subject of the inquiry is a remote terminal. The named connection can be either a connection entry that links towards the terminal-owning region (TOR), or an indirect connection that provides the netname of the TOR.

Otherwise this field is blank.

RNAME (*value*)
Displays the 4-character name by which this terminal is known in the system

in which it is defined (the TOR). RNAME applies only to terminals defined as remote; for others, blanks are displayed.

RNETNAME (*value***)**
Displays the 8-character netname of the owning TOR, if the subject of the inquiry is a remote terminal. If this field is blank and the terminal is remote, the system named in the REMOTESYSTEM field has not been installed, and no value was specified for the REMOTESYSNET option when the terminal was defined.

SERVSTATUS
Displays whether the terminal is available for use. The values are as follows:
INSERVICE
The terminal is available for use. For z/OS Communications Server, INSERVICE means that the terminal can be ACQUIRED. For IRC sessions, INSERVICE means that the connection to the MRO partner is INSERVICE.
OUTSERVICE
The terminal is not available for use. Setting a terminal OUTSERVICE means that the terminal can no longer be used by transactions. If PURGE or FORCEPURGE is also specified, any transaction that is using the terminal is terminated abnormally. If PURGE or FORCEPURGE is not specified, the transaction can terminate normally, but no further transactions can use the terminal. For z/OS Communications Server, setting a terminal OUTSERVICE also causes it to be released and the operator to be signed off, either immediately or when the current transaction has is completed.

For IRC sessions, OUTSERVICE means that the connection to the MRO partner is OUTSERVICE.

In an LU6.1 ISC session, the connection is set OUTSERVICE if there are no allocatable sessions left.

For physical terminals and LU6.1 sessions, you can reset this value by overtyping it with a different value.

TASK (*value***)**
Displays the task number of the transaction that is running on this terminal.

TERMINAL (*value***)**
Displays a 4-character terminal identifier (1 - 4 characters) as specified in an installed terminal definition.

TERMSTATUS (*z/OS Communications Server only***)**
Displays whether CICS is in session with the logical unit represented by this terminal. The values are as follows:
ACQUIRED
CICS is in session with the logical unit represented by the terminal.
RELEASED
CICS is not in session with the logical unit represented by the terminal. If you set this option to RELEASED, a session is terminated immediately if you also specify the PURGE option, otherwise the session is terminated when the current active transaction finishes.

For physical terminals and LU6.1 sessions, you can reset this value by overtyping it with a different value or with the following:
COLDACQ
This is a special form of ACQUIRED, where no resynchronization is required. If the previous session abended, the use of COLDACQ

overrides CICS integrity control. This might lead to integrity problems. Also check the CSMT log for an activity keypoint after the restart of a session following a CICS failure. If there is no activity keypoint, issue COLDACQ again after the next emergency restart.

TRANSACTION (*value***)**
Displays a 4-character string that indicates the name of the transaction currently being processed with this terminal as its principal facility or as a secondary facility.

TTISTATUS
Displays whether the terminal can be used by the transactions that are initiated from this terminal. The values are as follows:
TTI This terminal can be used by transactions.
NOTTI
This terminal cannot be used by transactions.

You can reset this value by overtyping it with a different value.

A terminal cannot be defined with both NOATI and NOTTI.

USERID (*data-area***)**
Returns the 8-character identifier of the user signed on at this terminal or session.

If there is no signed-on user, the default userid, as specified in the DFLTUSER system initialization parameter, is returned.

CEMT INQUIRE PARTNER

Retrieve information about the partner table.

Description

The **INQUIRE PARTNER** command returns information from the partner table (side information table) installed in your system. There is no SET PARTNER function. However, you can use the INQUIRE PARTNER screen to type a `D` in the left margin by the partner name to discard (the **CEMT DISCARD PARTNER** function).

For information on partners, see PARTNER resources.

Input

Press the Clear key to clear the screen. There are two ways of starting this transaction:

- Type `CEMT INQUIRE PARTNER` (or suitable abbreviations for the keywords). The resulting display lists the current status.
- Type `CEMT INQUIRE PARTNER` (or suitable abbreviations for the keywords), followed by the attributes that are necessary to limit the range of information that you require.

(*value***)**
Specifies the 8-character partner name as specified in the installed partner definition.

ALL
The default.

Sample screen

```
I PART
STATUS:  RESULTS
  Par(IST1ISC2)               Netn(IYMVM2  ) Pro(DFHCICSA)
     Ltp(04) Tpn(IST1                      )
  Par(IST2ISC2)               Netn(IYMVM2  ) Pro(DFHCICSA)
    Ltp(04) Tpn(IST2                      )
```

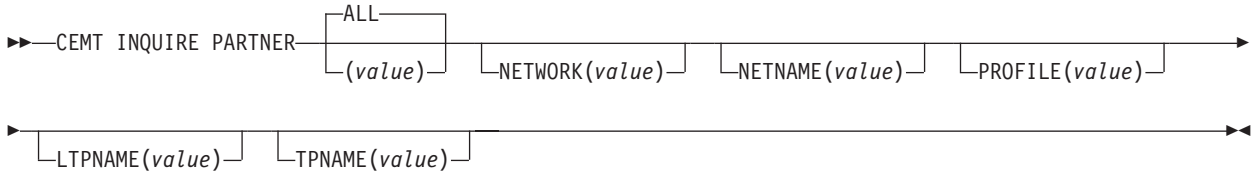
Figure 86. CEMT INQUIRE PARTNER screen

If you place the cursor against a specific entry in the list and press ENTER, CICS displays an expanded format as shown in Figure 87.

```
I PART
RESULT
  Partner(IST1ISC2)
  Network()
  Netname(IYMVM2)
  Profile(DFHCICSA)
  Ltpname(04)
  Tpname(IST1)
```

Figure 87. The expanded display of an individual entry

CEMT INQUIRE PARTNER



Displayed fields

LTPNAME(*value*)
Displays the length of the name of the remote transaction program (TPNAME), in bytes.

NETNAME(*value*)
Displays the 8-character name of the node at which the partner is located.

NETWORK(*value*)
Displays the 8-character name of the network in which the partner is located.

PARTNER(*value*)
Indicates that this panel relates to a PARTNER inquiry and displays the 8-character partner name as specified in the installed partner definition.

PROFILE(*value*)
Displays the 8-character name of the CICS profile that is associated with the partner.

TPNAME(*value*)
Displays the 64-character name of the remote transaction program.

Note: Nondisplayable characters appear as periods. You can use the PF2 key on the expanded panel to see the hexadecimal value.

CEMT INQUIRE PIPELINE

Retrieve information about an installed pipeline.

In the CICS Explorer, the Pipelines operations view provides a functional equivalent to this command.

Description

Use the **INQUIRE PIPELINE** command to retrieve information about an installed pipeline.

The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT INQUIRE PIPELINE` (or suitable abbreviations for the keywords). You get a screen that lists the current status.
- Type `CEMT INQUIRE PIPELINE` (or suitable abbreviations for the keywords), followed by the attributes that are necessary to limit the range of information that you require. For example, if you enter `cemt i pi en`, the resulting display shows the details of only those process-types that are enabled.

You can change various attributes in the following ways:

- Overtype your changes on the INQUIRE screen after tabbing to the appropriate field.
- Use the **CEMT SET PIPELINE** command.

ALL
The default. Information about all pipelines is displayed, unless you specify a selection of pipelines to be queried.

(value)
The 1- to 8-character name of an installed PIPELINE definition.

Displayed fields

CHANGEAGENT(*value***)**
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

CIDDOMAIN*(value)*
Displays the name of the domain that is used to generate MIME content-ID values that identify binary attachments.

CONFIGFILE*(value)*
Displays the name of the pipeline configuration file associated with the PIPELINE resource. The name can be up to 255 characters long.

DEFINESOURCE(*value***)**
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

ENABLESTATUS
Displays the status of the pipeline:

ENABLED
The pipeline is ready for use.

DISABLED
The pipeline is not processing requests, and cannot accept new work. It might have failed to initialize or might have been explicitly disabled.

ENABLING
The pipeline is being initialized; it is not yet ready to accept work.

DISABLING
The pipeline is quiescing before entering DISABLED state. It is not accepting new work, but is allowing current work to complete.

DISCARDING
A DISCARD command has been issued for the pipeline. The pipeline is quiescing before being discarded. It is not accepting new work, but is allowing current work to complete.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

MODE*(value)*
Displays the operating mode of the pipeline.

PROVIDER
CICS is using the pipeline as a service provider of web services.

REQUESTER
CICS is using the pipeline as a service requester of web services.

UNKNOWN
The operating mode of the pipeline cannot be determined.

MTOMNOXOPST*(value)*
Displays the status of the pipeline for sending outbound messages in MIME format when binary attachments are not present.

MTOMNOXOP
Outbound messages are sent in MIME format, even when there are no binary attachments present.

NOMTOMNOXOP
Outbound messages are sent in MIME format only when there are binary attachments present.

MTOMST*(value)*
Displays the status of the MTOM handler in the pipeline.

MTOM
The MTOM handler is enabled in the pipeline.

NOMTOM
The MTOM handler is not enabled in the pipeline.

RESPWAIT *(value)*
Displays the number of seconds that an application program waits for an

optional message from a remote web service. If no value is displayed, the default timeout value of the transport protocol is being used.

- The default timeout value for HTTP is 10 seconds.
- The default timeout value for WebSphere MQ is 60 seconds.

SENDMTOMST*(value)*
Displays the status of the pipeline for sending outbound messages in MIME format.

NOSENDMTOM
Outbound messages are never sent in MIME format.

SAMESENDMTOM
Outbound messages are sent in MIME format only when the inbound message is in MIME format.

SENDMTOM
Outbound messages are always sent in MIME format.

SHELF*(value)*
Displays the name of the *shelf directory*. The name can be up to 255 characters long.

SOAPLEVEL*(value)*
Displays the level of SOAP that is supported in the pipeline. The SOAP level can be 1.1 or 1.2. If the pipeline is not being used for SOAP messages, a value of NOTSOAP is displayed.

WSDIR*(value)*
Displays the name of the *web service binding directory* (also known as the *pickup directory*). The name can be up to 255 characters long.

XOPDIRECTST*(value)*
Displays the status of the pipeline for handling XOP documents and binary attachments in direct or compatibility mode.

XOPDIRECT
The pipeline is processing XOP documents and binary attachments in direct mode.

NOXOPDIRECT
The pipeline is processing XOP documents and binary attachments in compatibility mode.

XOPSUPPORTST*(value)*
Displays the status of the application handler for processing XOP documents and binary attachments directly.

XOPSUPPORT
The application handler supports the direct handling of XOP documents and binary attachments.

NOXOPSUPPORT
The application handler does not support the direct handling of XOP documents and binary attachments.

CEMT INQUIRE PROCESSTYPE

Retrieve information about a CICS business transaction services process-type.

In the CICS Explorer, the Processtypes operations view provides a functional equivalent to this command.

Description

The **INQUIRE PROCESSTYPE** command returns information about the BTS PROCESSTYPE definitions installed on this CICS region. It shows the current state of audit logging for each displayed process-type.

The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT INQUIRE PROCESSTYPE` (or suitable abbreviations for the keywords). The resulting display lists the current status.
- Type `CEMT INQUIRE PROCESSTYPE` (or suitable abbreviations for the keywords), followed by the attributes that are necessary to limit the range of information that you require. For example, if you enter `cemt i proc en`, the resulting display shows the details of only those process-types that are enabled.

You can change various attributes in the following ways:

- Overtype your changes on the INQUIRE screen after tabbing to the appropriate field.
- Use the CEMT SET PROCESSTYPE command.

ALL
The default. Information about all process-types is displayed, unless you specify a selection of process-types to be queried.

(value)
The name (1 - 8 characters) of one or more PROCESSTYPE definitions installed in the process-type table (PTT).

Sample screen

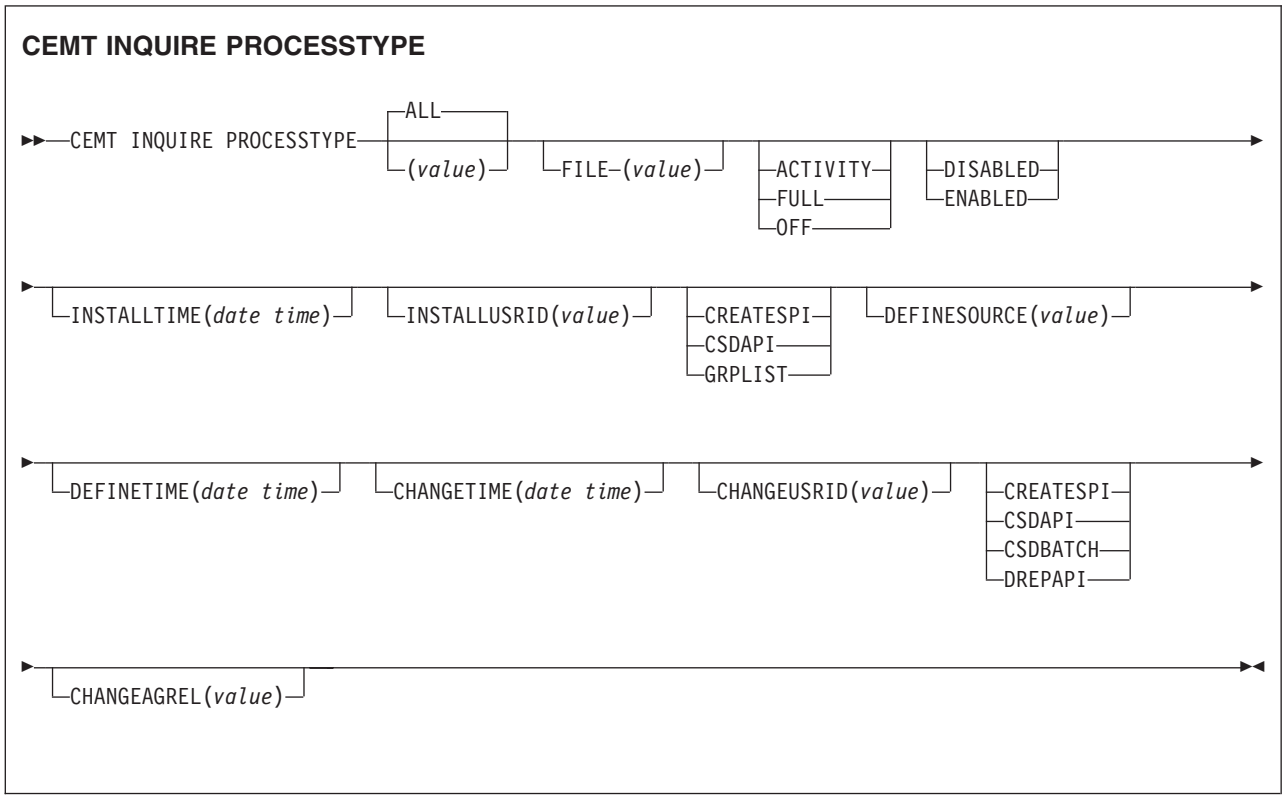
```
I PROC
STATUS:   RESULTS - OVERTYPE TO MODIFY
 Pro(PROCESSTYPE12  ) Fil(FILE12  ) Aud(ADTLOG12) Pro Ena
 Pro(PROCESSTYPE13  ) Fil(FILE13  ) Aud(ADTLOG12) Off Ena
 Pro(PTYPE2B        ) Fil(FILE2B  ) Aud(DFHJ2B  ) Ful Ena
 Pro(PTYPE39        ) Fil(FILE39  ) Aud(DFHJ39  ) Off Ena
 Pro(SALESTYPE1     ) Fil(SALESF1 ) Aud(PLOG51  ) Off Dis
 Pro(SALESTYPE4     ) Fil(SALESF4 ) Aud(PLOG51  ) Act Ena
 Pro(SALESTYPE6     ) Fil(SALESF6 ) Aud(PLOG51  ) Off Ena
```

Figure 88. CEMT INQUIRE PROCESSTYPE screen

If you place the cursor against a specific entry in the list and press ENTER, CICS displays an expanded format as shown in Figure 89 on page 284.

```
I PROC
STATUS:   RESULTS - OVERTYPE TO MODIFY
  Processtype(SALESTYPE4     )
  File(SALESF4 )
  Enablestatus( Enabled )
  Auditlog(PLOG51  )
  Auditlevel(Activity )
```

Figure 89. The expanded display of an individual entry



Displayed fields

AUDITLEVEL
Displays the level of audit logging currently active for processes of this type. The values are as follows:

ACTIVITY
Activity-level auditing. Audit records are written from:
1. The process audit points
2. The activity primary audit points.

FULL Full auditing. Audit records are written from:
1. The process audit points
2. The activity primary *and* secondary audit points.

OFF No audit trail records are written.

PROCESS
Process-level auditing. Audit records are written from the process audit points only.

For details of the records that are written from the process, activity primary, and activity secondary audit points, see the *CICS Business Transaction Services*.

AUDITLOG*(value)*
Displays the 8-character name of the CICS journal used as the audit log for processes of this type.

CHANGEAGENT(*value***)**
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

DEFINESOURCE(*value***)**
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

ENABLESTATUS
Displays whether new processes of this type can be created. The values are as follows:

DISABLED
The installed definition of the process-type is disabled. New processes of this type cannot be defined.

ENABLED
The installed definition of the process-type is enabled. New processes of this type can be defined.

FILE*(value)*
Displays the 8-character name of the CICS repository file on which the process and activity records for processes of this type are stored.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

PROCESSTYPE*(value)*
Indicates that this panel relates to a PROCESSTYPE inquiry and displays the 8-character name of a process-type.

CEMT INQUIRE PROFILE

Retrieve information about a transaction profile.

Description

The **INQUIRE PROFILE** command returns information about the profiles installed in your system. There is no SET PROFILE function. However, you can use the INQUIRE PROFILE screen to discard a profile. To do this, type a D in the left margin by the name of the profile to be discarded (the DISCARD PROFILE function).

The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key and type `CEMT INQUIRE PROFILE` (or suitable abbreviations for the keywords). The resulting display is similar to the following example.

(*value***)**
The name of a specific profile.

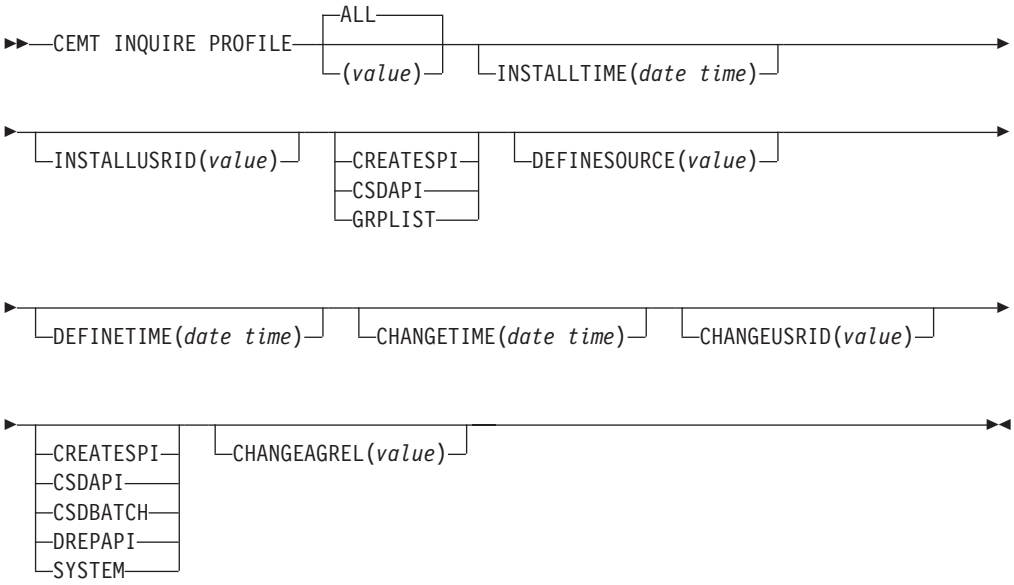
ALL
The default.

Sample screen

```
IN PROFILE
STATUS:  RESULTS
 Prof(DFHCICSA)
 Prof(DFHCICSE)
 Prof(DFHCICSF)
 Prof(DFHCICSR)
 Prof(DFHCICSS)
 Prof(DFHCICST)
 Prof(DFHCICSV)
```

Figure 90. CEMT INQUIRE PROFILE screen

CEMT INQUIRE PROFILE



Displayed fields

CHANGEAGENT(*value*)
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

SYSTEM
The resource definition was last changed by the CICS or CICSPlex system.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

DEFINESOURCE(*value***)**
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

PROFILE(*value***)**
Indicates that this panel relates to a PROFILE inquiry and displays the 8-character name of a profile installed in your system.

CEMT INQUIRE PROGRAM

Retrieve information about programs, maps, and partition sets.

In the CICS Explorer, the Programs operations view provides a functional equivalent to this command.

Description

INQUIRE PROGRAM returns information about the programs, map sets, and partition sets that are defined to your system. Java programs that run in a JVM return a subset of values.

The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key to clear the screen. Start this transaction in one of two ways:

- Type `CEMT INQUIRE PROGRAM`. You get a display that lists the status.
- Type `CEMT INQUIRE PROGRAM` followed by as many of the other attributes that are necessary to limit the range of information that you require. For example, if you enter `cemt i prog c e`, the resulting display shows you the details of only those programs that are written in C language and are enabled.

You can tab to the highlighted or blank fields and type over them with the required values.

(*value***)**
A 1 - 8 character program identifier which is the name of a specific program entry in the table of installed program definitions. If you omit the program name, the ALL option is assumed by default. You can also use the wildcard character * to return a selection of programs; for example, entering `CEMT INQUIRE PROGRAM(DFH*)` returns all programs that begin with DFH.

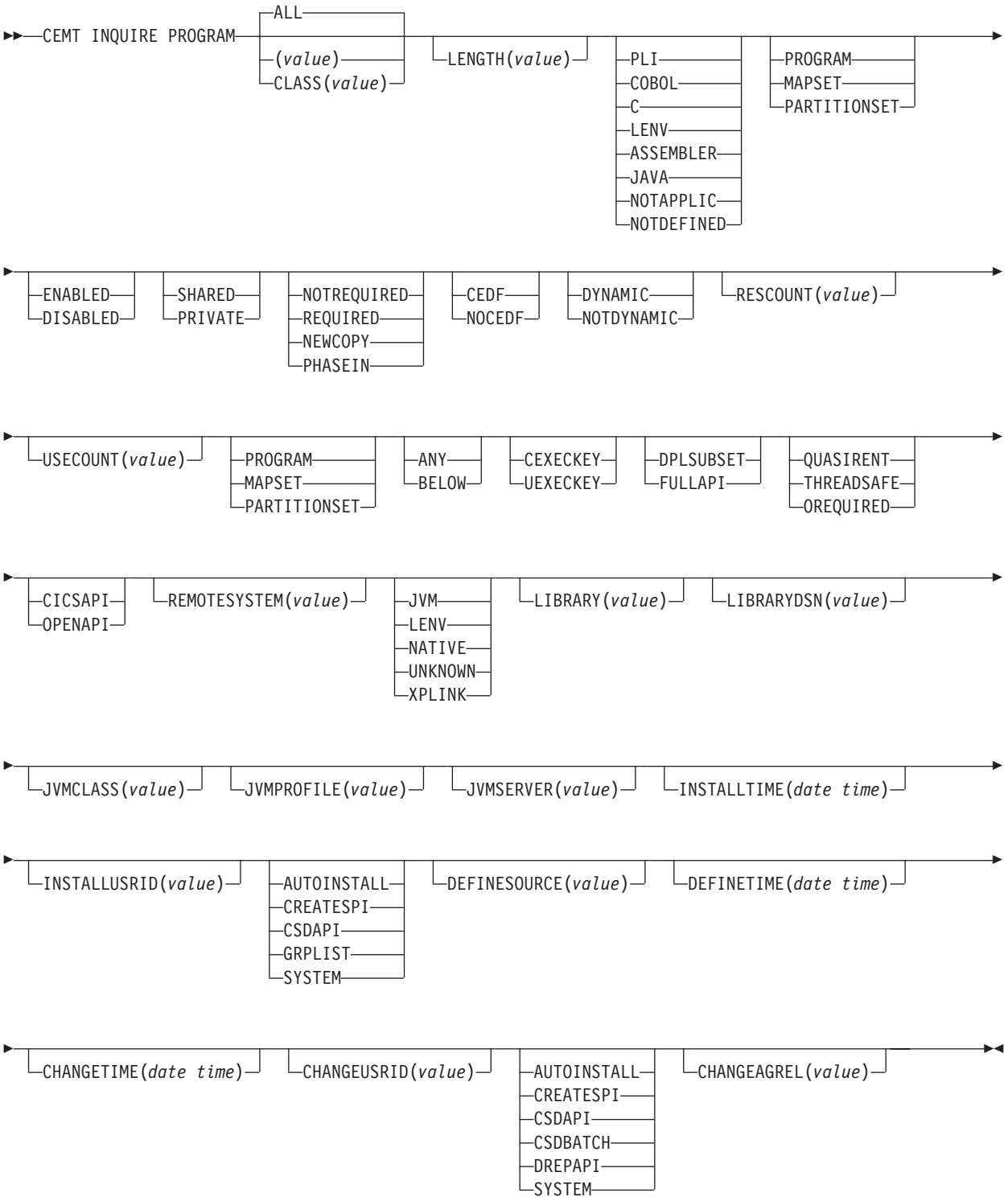
You cannot specify the CLASS option and a program name.

ALL
The default.

CLASS(*value***)**
The 2-character suffix of a program list table (PLT). Use this option to inquire on those programs defined in the specified PLT. Unlike a shutdown PLT, it is not mandatory to define a program definition for a startup PLT. However, if you want to use the CLASS option, you must define and install a program resource definition for the PLT specified by *clasid*.

If the *clasid* specified is the suffix of a PLT that does not have a program definition, the **INQUIRE PROGRAM CLASS***(clasid)* command returns a `CLASS NOT FOUND` message.

CEMT INQUIRE PROGRAM



Displayed fields

APIST
Displays the API attribute of the installed program definition.

CICSAPI
The program is restricted to use of the CICS permitted application programming interfaces only.

If the program is defined with CONCURRENCY(QUASIRENT) it always runs on the quasi-reentrant (QR) TCB. If the program is defined with CONCURRENCY(THREADSAFE) it runs on whichever TCB is in use by CICS at the time that is determined as suitable, and if the program is defined with CONCURRENCY(REQUIRED) it always runs on an L8 open TCB.

OPENAPI
The program is not restricted to the CICS permitted application programming interfaces.

CICS runs the program on its own L8 or L9 mode open TCB, depending on the value of the EXECKEY attribute. If the program is defined with CONCURRENCY(REQUIRED), CICS uses an L9 TCB if EXECKEY(USER) is set and an L8 TCB if EXECKEY(CICS) is set. If, when running a CICS command, CICS requires a switch to QR TCB, it returns to the open TCB before handing control back to the application program. OPENAPI requires the program to be coded to threadsafe standards.

CEDFSTATUS
Displays the action that the execution diagnostic facility (EDF) is to take for this program when the program runs under EDF.

CEDF
When EDF is active, CEDF initiation and termination screens are shown by CEDF while this program is running. Other screens are also shown unless the program was translated using the NOEDF translator option.

NOCEDF
All CEDF activities, including initiation and termination screens, stop while this program is being processed.

CHANGEAGENT(*value*)
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

AUTOINSTALL
The resource was autoinstalled.

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

SYSTEM
The resource definition was last changed by the CICS or CICSPlex system.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

CONCURRENCY
Displays the concurrency attribute of the specified program.
QUASIRENT
The program is defined as not threadsafe, and requires the automatic serialization provided by QR TCB.
THREADSAFE
The program is defined as being threadsafe, using appropriate serialization techniques to access shared resources. It does not need the automatic serialization provided by the CICS QR TCB, and can safely run on an open TCB.
OREQUIRED
The program is defined as threadsafe, and must run on an open TCB. The type of open TCB used depends on the API setting.

A value of CONCURRENCY(OREQUIRED) is equivalent to CONCURRENCY(REQUIRED) on the program definition. CEMT uses a value of OREQUIRED to distinguish it from a value of REQUIRED on the option COPYSTATUS. You can use the filter OREQUIRED to return all programs with the concurrency setting of REQUIRED.

COPYSTATUS
Displays the load status of the specified program.
NOTREQUIRED
The program status is such that you do not have to specify **CEMT SET PROGRAM NEWCOPY** or **CEMT SET PROGRAM PHASEIN** to make the program available. If the program is already available for use and you have put a new version in the DFHRPL or a dynamic LIBRARY concatenation, issue a NEWCOPY or PHASEIN command to replace the current version.
REQUIRED
A search for the program failed during a load operation, and the program has been marked as not loadable to avoid further load attempts.

You must specify **CEMT SET PROGRAM NEWCOPY** or **CEMT SET PROGRAM PHASEIN** to enable the program to be loaded. Ensure that a copy of the program is in the search order of LIBRARY concatenations, including DFHRPL and dynamic libraries.

You can reset this value by typing over it using the following values:
NEWCOPY
CICS uses a new copy of the program if all the transactions currently using the program have finished (RESCOUNT equal to zero). If

RESCOUNT is not zero, NEWCOPY fails. CICS loads the new version either from the search order of LIBRARY concatenations (including DFHRPL and dynamic LIBRARYs) or uses an LPA-resident version, depending on the PRIVATE or SHARED options. PRIVATE is the default setting.

You cannot specify NEWCOPY for a program specified with the HOLD option, or for a Java program that runs in a JVM.

PHASEIN
CICS uses a new copy of the program now for all new transaction requests. CICS continues to use the old copy for all currently running transactions until they have finished (RESCOUNT equal to zero). CICS loads the new version either from the search order of LIBRARY concatenations (including DFHRPL and dynamic LIBRARYs) or uses an LPA-resident version, depending on the PRIVATE or SHARED options. PRIVATE is the default setting.

You cannot specify PHASEIN for a program specified with the HOLD option, or for a Java program that runs in a JVM.
PHASEIN performs a REFRESH PROGRAM function to inform the loader domain that a new version of the program has been cataloged and that this version of the named program is used in all future ACQUIRE requests.
Until the program goes to a zero RESCOUNT and a subsequent **ACQUIRE PROGRAM** has been performed, **CEMT INQUIRE PROGRAM** returns information about the first copy of the program.

DATALOC
Displays whether the program can accept data addresses higher than 16 MB. The values are as follows:

ANY The program can handle 31-bit addresses. The address of the data can be above or below 16 MB. CICS returns an address above 16 MB if possible.

BELOW
The program can handle only 24-bit addresses and must therefore be given addresses of data located below 16 MB. If required, CICS copies data below 16 MB before passing its address to the application program.

DEFINESOURCE(*value***)**
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

DYNAMSTATUS
Displays whether, if the program is the subject of a program-link request, the request can be dynamically routed.

DYNAMIC
If the program is the subject of a program-link request, the CICS dynamic routing program is called. Providing that a remote server region is not named explicitly on the SYSID option of the **EXEC CICS LINK** command, the routing program can route the request to the region on which the program is to run.

NOTDYNAMIC
If the program is the subject of a program-link request, the dynamic routing program is not called.

For a distributed program link (DPL) request, the server region on which the program is to run must be specified explicitly on the REMOTESYSTEM option of the PROGRAM definition or on the SYSID option of the **EXEC CICS LINK** command; otherwise, it defaults to the local region.

For information about the dynamic routing of DPL requests, see in the *CICS Intercommunication Guide*.

EXECKEY
Displays the access key in which the program is running.

CEXECKEY
The program runs in CICS access key, and has read and write access to both CICS-key and user-key storage.

UEXECKEY
The program runs in user access key, and has write access to user-key storage, but read-only access to CICS-key storage.

EXECUTIONSET
Displays whether the program is restricted to the API subset as for a distributed program link request. The values are as follows:

DPLSUBSET
The program is restricted to the DPL API subset, as for a distributed program link request, when it runs in the local CICS region.

A program is always restricted to the DPL subset when it is called in a remote region with a DPL request, even if this option is not specified.

FULLAPI
The program is not restricted to the DPL subset of the CICS API when it runs in the local CICS region and can use the full API.

A program is always restricted to the DPL subset when it is called in a remote region with a DPL request, regardless of this option.

You can reset this value by typing over it with a different value.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

AUTOINSTALL
The resource was autoinstalled.

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

SYSTEM
The resource was installed by the CICS or CICSPlex SM system.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the

date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value*)
Displays the 8-character user ID that installed the resource.

JVMCLASS
Displays the name of the main class in the Java program or OSGi service, as specified in the PROGRAM definition, up to a maximum of 255 characters.

JVMPROFILE
Displays the name of the JVM profile that is used for the pooled JVM in which this Java program runs.

JVMSERVER
Displays the name of the JVMSERVER resource that is used to run this OSGi service in a JVM server.

LANGUAGE
Displays the language in which the program is written.

PLI, Cobol, C, Lenv, or Assembler
Supported languages. **Lenv** Means that the module, whatever its language, uses multi-language support or was compiled with a Language Environment-conforming compiler.

JAVA The program is written in Java.

NOTAPPLIC
The language is not specified in the resource definition, and is not loaded.

NOTDEFINED
The language in which the program is written is not defined.

LENGTH(*value*)
Displays the size of the program in bytes. The value returned is `0000000000` if the program has not been loaded during this CICS run. No value is returned if the module is a remote program or a Java program that runs in a JVM.

LIBRARY(*value*)
Displays the 8-character name of the LIBRARY from which the program is currently loaded.

Note:

- If the program was loaded from an installed LIBRARY resource, the library and library data set names are returned.
- If the program was loaded from a LIBRARY resource that is disabled, the library name is returned but the library data set name is blank.
- If the program was loaded from a LIBRARY resource that is discarded, both library and library data set name are blank.

LIBRARYDSN(*value*)
Displays the 44-character name of the data set within the LIBRARY from which the program is currently loaded.

Note:

- If the program was loaded from an installed LIBRARY resource, the library and library data set names are returned.
- If the program was loaded from a LIBRARY resource that is disabled, the library name is returned but the library data set name is blank.

- If the program was loaded from a LIBRARY resource that is discarded, both library and library data set name are blank.

PROGRAM(*value*)
Indicates that this panel relates to a PROGRAM inquiry and displays an 8-character program identifier (1 - 8 characters), which is the name of a specific program entry in the table of installed program definitions.

PROGTYPE
Displays the type of program:
PROGRAM
The entry is defined in the CICS system definition as a program.
MAPSET
The entry is defined in the CICS system definition as a map set.
PARTITIONSET
The entry is defined in the CICS system definition as a partition set.

REMOTESYSTEM(*value*)
Displays the 4-character name of the remote system in which the program is to run.

RESCOUNT(*value*)
Displays a 3-character string identifying the number of separate invocations of this program that are taking place at the time of this inquiry. No value is returned if the module is a remote program or a Java program that runs in a JVM.

This number is not necessarily the same as the number of times the module has been loaded from DFHRPL or a dynamic LIBRARY concatenation.

RUNTIME
Displays information about the runtime environment of the program.
JVM The program is a Java program that runs in a Java Virtual Machine (JVM).
Lenv The program is running with Language Environment runtime support.
NATIVE
The program is running with its native language runtime support (not Language Environment).
NOTKNOWN
The runtime environment is not known because the program was not loaded by CICS. The source language, which dictates the runtime environment to be used, cannot be deduced.
XPLINK
The program is a C or C++ program which was compiled using the XPLINK option.

SHARESTATUS
Displays, when the next NEWCOPY or PHASEIN request is received, where the next new copy of the program is to be loaded from.
SHARED
CICS is using a shared copy of the program from the link pack area (LPA). If no LPA version is available, the program is loaded from DFHRPL or a dynamic LIBRARY concatenation.
PRIVATE
The next new copy of the program is to be loaded from DFHRPL or a dynamic LIBRARY concatenation.

You can reset this value by typing over it with a different value.

STATUS
Displays whether the program is available for use using of the following values:
ENABLED
The program is available for use.
DISABLED
The program is unavailable for use.

Programs beginning with "DFH" cannot be disabled because these characters are reserved for use by CICS.

You can reset this value by typing over it with a different value.

USECOUNT(*value***)**
Displays a 10-character string identifying the total number of times the program has run since the start of the current CICS session. The use count is provided for all modules including Java programs, except for remote programs. No value is returned if the program is remote.

CEMT INQUIRE REQUESTMODEL

Retrieves information about a REQUESTMODEL definition.

Description

The **INQUIRE REQUESTMODEL** command returns information about any currently installed request models.

The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key to clear the screen, and type `CEMT INQUIRE REQUESTMODEL` (or suitable abbreviations for the keywords). The resulting display lists the names and status of any request models.

(*value***)**
Displays the 4-character field that contains the (possibly generic) name of the destination CORBAServer for this REQUESTMODEL.

ALL
The default.

Sample screen

```
I RE
STATUS:  RESULTS - OVERTYPE TO MODIFY
 Req(CFAC    ) Corbas(IIOP) Tra(CICF) Cor    Con Ign
 Req(III     ) Corbas(IIII) Tra(CIRP) Cor    Con Res
    Mod(*                         ) Int(*                       )


                                          SYSID=HA61 APPLID=CICSHA61
 RESPONSE: NORMAL                       TIME:  15.54.22  DATE: 09.03.98
PF 1 HELP       3 END       5 VAR        7 SBH 8 SFH 9 MSG 10 SB 11 SF
```

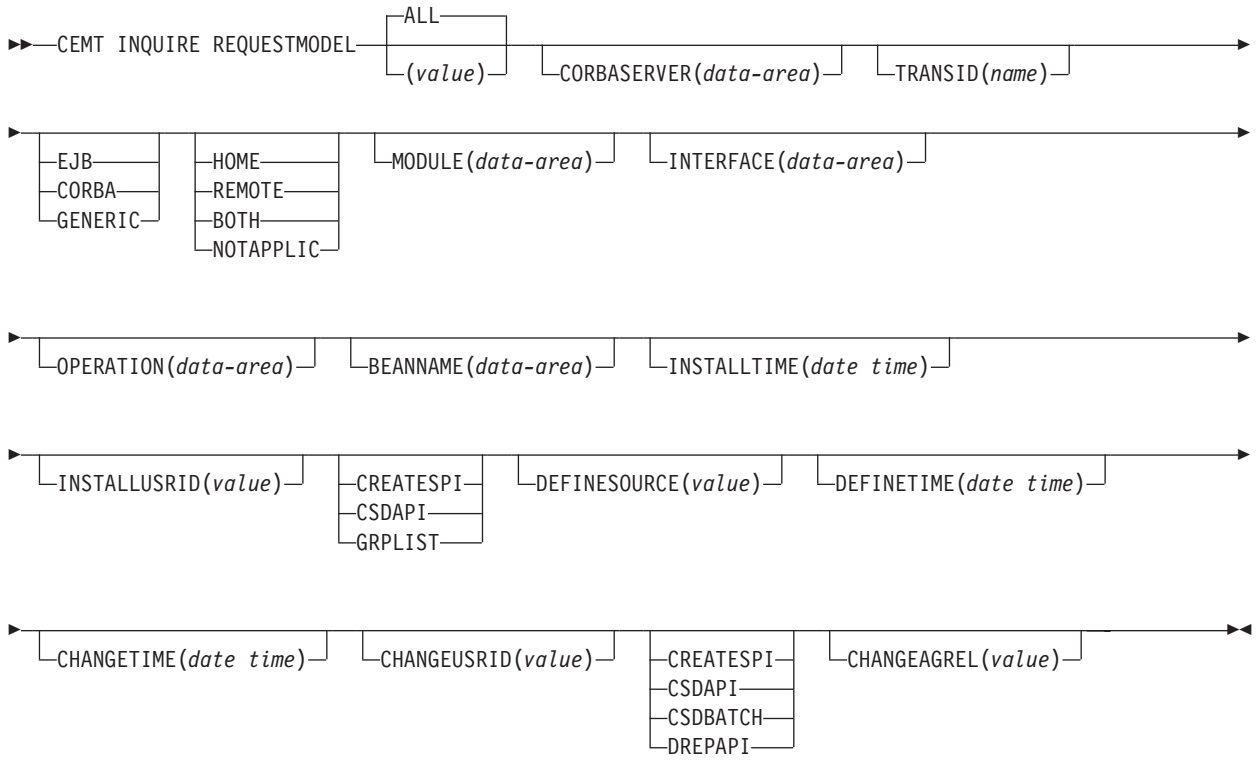
Figure 91. CEMT INQUIRE REQUESTMODEL screen

If you place the cursor against a specific entry in the list and press ENTER, CICS displays an expanded format.

```
I RE
RESULT - OVERTYPE TO MODIFY
  All
  Requestmodel(DFHGFACT)
  Corbaserver
  Transid
  Type
  Intfacetype
  Module
  Interface
  Operation
  Beanname
```

Figure 92. The expanded display of an individual entry

CEMT INQUIRE REQUESTMODEL



Displayed fields

(*value***)**
Displays the 4-character field that contains the (possibly generic) name of the destination CORBAServer for this REQUESTMODEL.

BEANNAME*(DATA-AREA)*
Displays the 240-character field that contains the (possibly generic) bean name that matches the name of an enterprise bean in an XML deployment descriptor. This field is blank if the TYPE attribute is CORBA.

CHANGEAGENT(*value***)**
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

CORBASERVER*(DATA-AREA)*
Displays the 4-character field that contains the (possibly generic) name of the destination CORBAServer for this REQUESTMODEL.

DEFINESOURCE(*value***)**
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

INTERFACE*(DATA-AREA)*
Displays the 255-character field that contains the (possibly generic) pattern matching the interface name for this REQUESTMODEL. This field is blank if the TYPE attribute is EJB.

INTFACETYPE
Displays a value that indicates the Java interface type for this REQUESTMODEL. The values are as follows:

HOME
Specifies that this is the home interface for the bean.

REMOTE
Specifies that this is the component interface for the bean.

BOTH Matches both the home and component interfaces for the bean.

NOTAPPLIC
Returned if the TYPE attribute is CORBA.

MODULE*(DATA-AREA)*
Displays the 255-character field that contains the (possibly generic) pattern that defines the name scope of the interface and operation values for this REQUESTMODEL. This field is blank if the TYPE attribute is EJB.

OPERATION*(DATA-AREA)*
Displays the 31-character field that contains the pattern that matches the IDL operation name for this REQUESTMODEL.

REQUESTMODEL (*value***)**
Displays the 8-character request model name.

TRANSID *(name)*
Displays the 4-character name of the CICS transaction execute when a request that matches the specification of the REQUESTMODEL is received.

TYPE
Displays a value that indicates the type of the REQUESTMODEL. The values are as follows:

EJB Matches enterprise bean requests as specified by the EJB parameters.

CORBA
Matches CORBA requests as specified by the CORBA parameters.

GENERIC
Matches both enterprise bean and CORBA requests.

CEMT INQUIRE RRMS

Retrieve recoverable resource management services (RRMS) information.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

INQUIRE RRMS indicates whether CICS accepts inbound transactional external CICS interface (EXCI) work.

Input

Press the Clear key and type `CEMT INQUIRE RRMS` (or suitable abbreviations for the keywords).

The resulting display is similar to the following example.

Sample screen

```
IN RRMS
STATUS:  RESULTS
 Openstatus(Open)
```

Figure 93. CEMT INQUIRE RRMS screen

Displayed fields

OPENSTATUS
Displays a value that indicates the status of inbound transactional EXCI work. The values are:

CLOSED
CICS does not accept inbound transactional EXCI work.

OPEN
CICS does accept inbound transactional EXCI work.

UNAVAILABLE
CICS is initialized without RRMS.

CEMT INQUIRE STATISTICS

Retrieve statistics information.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

The **INQUIRE STATISTICS** command shows the interval during which statistics are accumulated, the end-of-day time for shutdown (or logical end-of-day) statistics, the time at which statistics are next recorded (that is, written to an SMF data set), and whether statistics recording is currently on or off.

For details of how to print statistics using the CICS-supplied utility program DFHSTUP, see the *CICS Operations and Utilities Guide*.

Input

Press the Clear key and type `CEMT INQUIRE STATISTICS` (or suitable abbreviations for the keywords).

The resulting display is similar to the following example. To change any of the values shown, tab to the highlighted or blank fields and overtype them as required.

Sample screen

```
IN STATI
STATUS:  RESULTS - OVERTYPE TO MODIFY
 Sta On          Int( 030000 ) End( 000000 ) Nex(150000)
```

Figure 94. CEMT INQUIRE STATISTICS screen

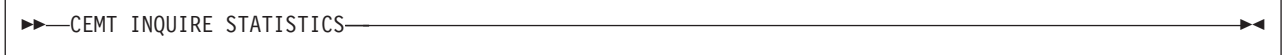
Note: When a value does not apply, or is negative (the value begins with No), the fields on the screen are blank. To modify these fields, locate them by tabbing (they appear in the same sequence as in the expanded format), and overtype with input valid for that field. You might find it more convenient to use the expanded format when setting one of these values.

If you place the cursor against the result and press ENTER, CICS displays an expanded format, as shown in Figure 95.

```
IN STAT
RESULT - OVERTYPE TO MODIFY
  Statistics
  Recording( Off )
  Record(              )
  Reset(              )
  Interval( 030000 )
  Endofday( 000000 )
  Nexttime(000000)
```

Figure 95. The expanded display

CEMT INQUIRE STATISTICS



Displayed fields

ENDOFDAY (*hhmmss*)
Displays the end-of-day time for recording statistics. At end-of-day, the current values of the CICS statistics counters are written out to an SMF data set and the statistics counters are reset.

End-of-day times must be in the range 000000 - 235959. The default end-of-day time is 12 midnight (000000).

INTERVAL (*hhmmss*)
Displays the length of time during which the statistics counters are incremented. At the end of each interval, the accumulated statistics are recorded and the statistics counters are reset.

You can reset this value by overtyping it with a different value.

Interval times must be in the range 000100 - 240000. The default interval value is 030000. You can change the interval value, but the change has no effect unless the statistics recording status is set ON. You can set the statistics

recording status ON or OFF by using the RECORDING option on this command, or by using the STATRCD system initialization parameter.

The arrival of the ENDOFDAY time always causes the current interval to be ended (possibly prematurely) and a new interval to be started. Only end-of-day statistics are recorded at the end-of-day time, even if it coincides exactly with the expiry of an interval.

Changing the ENDOFDAY value has an immediate effect on the times at which INTERVAL statistics are recorded. New interval expiry times are recalibrated in accordance with the new end-of-day time.

When you change the INTERVAL value or the ENDOFDAY value (and also when CICS is initialized), the length of the current (or first) interval is adjusted so that it expires after an integral number of intervals from the end-of-day time.

These rules are illustrated by the following example, where **I** indicates an interval recording, INTERVAL is set at 3-hourly intervals, CICS is initialized at 0815 hours, and **E** indicates the ENDOFDAY setting at 1700 hours.

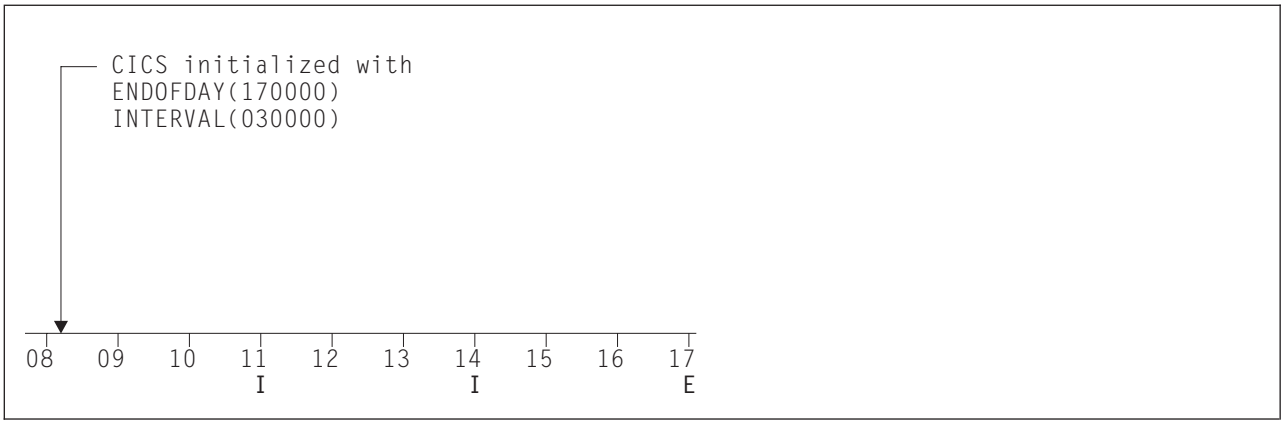


Figure 96. Changing the INTERVAL or ENDOFDAY value

Note: In this example, there would have been an interval at 0800 hours if CICS had been initialized earlier.

If you want your end-of-day recordings to cover 24 hours, set INTERVAL to 240000.

NEXTTIME (*hhmmss*)
Displays the time at which statistics are next recorded and reset. This is the expiry time of the current interval or the end-of-day time, whichever is earlier.

RECORD (input only field)
Specifies whether existing statistics are to be written to an SMF data set. The value is as follows:

RECORD
Existing statistics are written to an SMF data set. The statistics are those collected since the last ENDOFDAY or INTERVAL collection, or since the last time the RESETNOW function was used in this command or the PERFORM STATISTICS command.

Note:

1. When both Rec(ord) and Res(et) are specified, statistics counters are written to the SMF data set and reset. The statistics that you are given are those

collected since the last ENDOFDAY or INTERVAL collection, or since the last time the RESETNOW function was used in this or the PERFORM STATISTICS command.

2. RECORD RESET takes effect only when the recording status is changed.

RECORDING
Displays whether the recording of interval statistics is switched on or off. End-of-day statistics, requested statistics, and unsolicited statistics are always recorded, irrespective of the setting of the ON or OFF option. (Requested statistics are those called for by a CEMT PERFORM STATISTICS command or by an EXEC CICS PERFORM STATISTICS RECORD command.) The values are as follows:

ON Interval statistics are recorded. Counters are incremented as usual, and are reset at the expiry of each interval and at end-of-day.

OFF Interval statistics are not recorded. Counters are still incremented as usual, and end-of-day, requested, and unsolicited statistics are still recorded.

You can reset this value by overtyping it with a different value.

RESET (input only field)
Specifies whether the statistics counters are reset. The value is as follows:

RESET
The statistics counters are reset. The reset takes effect only if the recording option is changed from OFF to ON or from ON to OFF. For information about the effect of reset on each individual counter, see the description of the CICS statistics tables in the *CICS Performance Guide*.

The action, for each counter, is one of the following:
- Not reset
- Reset to zero
- Reset to 1
- Reset to current value (this applies to peak values)
- An exception to the above.

STATISTICS
Indicates that this panel relates to a STATISTICS inquiry.

CEMT INQUIRE STREAMNAME

Retrieve information about currently connected MVS log streams.

Description

You can use the **INQUIRE STREAMNAME** command to inquire on the currently connected MVS log streams.

Input

Press the Clear key to clear the screen. There are two ways to start this transaction:

- Type `CEMT INQUIRE STREAMNAME` (or suitable abbreviations for the keywords). The resulting display lists the current status.
- Type `CEMT INQUIRE STREAMNAME` (or suitable abbreviations for the keywords), followed by the attributes that are necessary to limit the range of information that you require. For example, if you enter `cemt i str fai sys`, the resulting display shows the details of only those stream names that are failed and that are the system log.

You can then tab to the highlighted or blank fields and overtype them with the required values.

(*value***)**
Specifies an MVS system logger log stream name. The name can be a specific LSN of up to 26 characters.

ALL
The default.

Sample screen

```
IN STRE
STATUS:  RESULTS
 Str(BELL.CICSHT61.DFHLOG      ) Use(0000001) Ok  Sys
```

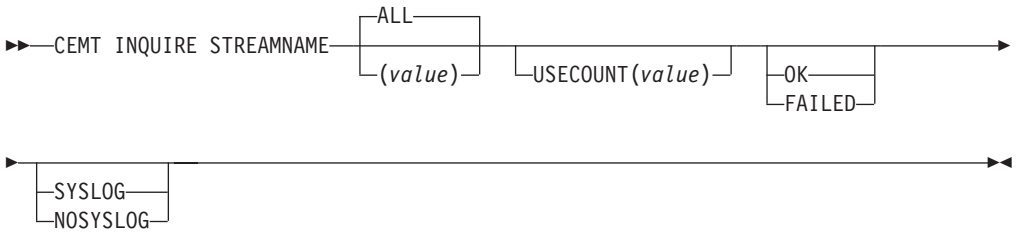
Figure 97. CEMT INQUIRE STREAMNAME screen

If you place the cursor against a specific entry in the list and press ENTER, CICS displays an expanded format, as shown in Figure 98.

```
IN STRE
RESULT
  Streamname(BELL.CICSHT61.DFHLOG)
  Usecount(0000001)
  Status(Ok)
  Systemlog(Syslog)
```

Figure 98. The expanded display of an individual entry

CEMT INQUIRE STREAMNAME



Displayed fields

STATUS
Displays the status of the log stream. The values are as follows:
OK No errors have been detected.
FAILED
The MVS system logger has detected a problem with the specified log stream.

STREAMNAME (*value***)**
Indicates that this panel relates to a STREAMNAME inquiry and displays an MVS system logger log stream name.

SYSTEMLOG
Displays whether this log stream is the system log. The values are as follows:
SYSLOG
The log stream is the system log.
NOSYSLOG
The log stream is not the system log.

USECOUNT (*value***)**
Displays the number of CICS journal names and forward recovery logs that are currently using the log stream.

The use count is always at least 1, because CICS does not maintain any information about a log stream that no longer has any users, in which case an INQUIRE STREAMNAME command returns an ERROR condition.

If the log stream name refers to the CICS system log, the use count is always 1. This is ture even when user application programs write recovery records to the CICS system log.

CEMT INQUIRE SYDUMPCODE

Retrieve information about the system dump code table.

Description

You can use the **INQUIRE SYDUMPCODE** command to see the current settings of the entries in the system dump table. Settings that begin with NO, for example NOSYSDUMP, are shown as blank.

The INQUIRE command shows the following information:

- The system dump code or message identifier
- Whether CICS produces a system dump
- Whether the system dump request is local or related
- Whether CICS is to shut down following the dump
- The current number of dumps produced
- The maximum number of dumps
- Whether a dump produced by CICS is eligible for suppression by the MVS Dump Analysis and Elimination (DAE) component.

A system dump code is the CICS message number with the DFH prefix removed. Typically, this leaves a 6-character code comprising 2 alphabetic characters and 4 numeric characters, for example AP0001. The command syntax permits up to 8 characters for the dump code and the dump codes you specify are padded with trailing spaces.

Each message has its own system default actions that you cannot necessarily override with the SET command. For example, when a serious error occurs, CICS shuts down even if you have asked for no shutdown. See *CICS Messages and Codes Vol 1* for information about the system action for each message.

Other system parameters can take priority over your dump table entries. For example, an EXEC CICS SET SYSTEM DUMPING NOSYSDUMP command, or a DUMP=NO system initialization parameter, inhibits system dumping. For information on using dumps, see the *CICS Problem Determination Guide*.

The INQUIRE SYDUMPCODE command returns entries that were added to the dump table by a user application program or CEMT command, and also entries that CICS added to the dump table through dump requests. Be aware that the entries that CICS added to the dump table are temporary, and they are not restored during a warm start, or during a cold or initial start. Entries added by a user application program or CEMT command are restored during a warm start, but not during a cold or initial start.

The transaction dump command **SET TRDUMPCODE** can also cause CICS to produce system dumps for the specified transaction dump code.

Input

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT INQUIRE SYDUMPCODE` (or suitable abbreviations for the keywords). The resulting display lists the current status.
- Type `CEMT INQUIRE SYDUMPCODE` (or suitable abbreviations for the keywords), followed by the attributes that are necessary to limit the range of information that you require. For example, if you enter `cemt i syd sy sh`, the resulting display shows the details of only those system dump codes for which a system dump is to be produced and CICS is to be shut down.

You can tab to the highlighted or blank fields and overtype them with the required values.

(*value***)**
An 8-character system dump code. See *CICS Messages and Codes Vol 1* for details of the codes.

ALL
The default.

Sample screen

```
I SYD
STATUS:  RESULTS - OVERTYPE TO MODIFY
 Syd(AP0001  ) Sys Loc     Max( 010 ) Cur(0000)     Dae
 Syd(AP0002  ) Sys Loc     Max( 999 ) Cur(0000)
```

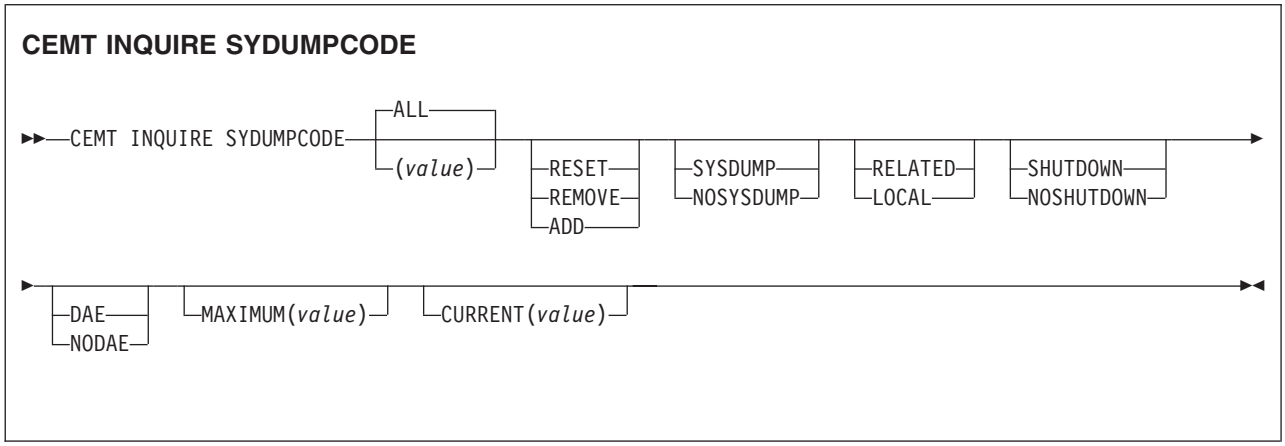
Figure 99. CEMT INQUIRE SYDUMPCODE screen

Note: When a value does not apply, or is negative (the value begins with No), the fields on the screen are blank. To modify these fields, locate them by tabbing (they appear in the same sequence as in the expanded format), and overtype with input valid for that field. You might find it more convenient to use the expanded format when setting one of these values.

If you place the cursor against the result and press ENTER, CICS displays an expanded format, as shown in Figure 100 on page 309.

```
I SYD
RESULT - OVERTYPE TO MODIFY
  Sydumpcode(AP0001)
  Sysdumping( Sysdump )
  Dumpscope( Local )
  Shutoption( Noshutdown )
  Maximum( 010 )
  Current(0000)
  Action(             )
  Daeoption( Dae )
```

Figure 100. The expanded display



Displayed fields

ACTION (*input only field***)**
Specifies the action to take with this system dump code. The values are as follows:

RESET
The current number of calls for this dump code is reset to zero.

REMOVE
This system dump code is removed from the system dump code table.

ADD This system dump code is added to the system dump code table.

CURRENT (*value***)**
Displays the number of dump calls that have been made for this dump code since it was last reset.

DAEOPTION
Displays whether a dump taken for CICS is eligible for suppression by the MVS dump analysis and elimination (DAE) component. The values are as follows:

DAE A system dump is eligible for suppression by the DAE component.

NODAE
A system dump is not suppressed by the DAE component. However, the SUPPRESS and SUPPRESSALL options in the ADYSETxx parmlib member might lead to dump suppression even though NODAE is set here. These options are controlled by the VRADAE and VRANODAE keys in the SDWA. For information about DAE, SUPPRESS, and SUPPRESSALL, see *z/OS MVS Diagnosis: Tools and Service Aids*.

You can reset this value by overtyping it with a different value.

DUMPSCOPE (*value*)
Displays whether a system dump request is sent to MVS images in the sysplex that are running XCF/MRO connected CICS regions related to the CICS region on which the dump is initiated.

Note: A related CICS region is one in which the unit of work identifiers, in the form of APPC tokens, of one or more tasks match those in the CICS region that issued the dump request.

RELATED
A system dump request is sent for this system dump code.

LOCAL
A system dump request is not sent to other MVS images in the sysplex for this system dump code.

You can reset this value by overtyping it with a different value.

MAXIMUM (*value*)
Displays the maximum number of dump calls for this system dump code that result in system dumps being taken. The value is in the range 0 - 999 and can be specified by using SYDUMAX in the SIT parameter. A value of 999 means the default, no limit.

You can reset this value by overtyping it with a different value.

SHUTOPTION
Displays whether CICS is shut down after an error that corresponds to this dump code occurs.

SHUTDOWN
CICS is shut down.

NOSHUTDOWN
CICS is not shut down.

You can reset this value by overtyping it with a different value.

SYDUMPCODE (*value*)
Indicates that this panel relates to a SYDUMPCODE inquiry and displays an 8-character system dump code. See *CICS Messages and Codes Vol 1* for details of the codes.

SYSDUMPING
Displays whether a system dump is produced for this system dump code. The values are as follows:

SYSDUMP
Produce a system dump for this system dump code.

NOSYSDUMP
Do not produce a system dump for this system dump code.

You can reset this value by overtyping it with a different value.

CEMT INQUIRE SYSTEM

Retrieve information about the CICS system.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

INQUIRE SYSTEM returns information about the CICS system.

Input

Press the Clear key to clear the screen, and type CEMT INQUIRE SYSTEM. The result is a display that lists the current status. You can then tab to the highlighted or blank fields and overtype them with the required values.

To change various attributes, you can:

- Overtype your changes on the INQUIRE screen after tabbing to the appropriate field (see Figure 101)
- Use the **CEMT SET SYSTEM** command.

Sample screen

```
 inq system
STATUS:  RESULTS - OVERTYPE TO MODIFY
  Aging( 32768 )                Progautoctlg( Ctlgmodify )
  Akp( 04000 )                  Progautoexit( DFHPGADX )
  Cicstslevel(030100)           Progautoinst( Autoactive )
  Cmdprotect(Nocmdprot)         Reentprotect(Reentprot)
  Db2conn()                     Release(0670 )
  Debugtool( Nodebug )          Runaway( 0005000 )
  Dfltuser(WALSH)               Scandelay( 0500 )
  Dsalimit( 07340032 )          Sdtran(CESD)
  Dsrtprogram( NONE     )       Sosabovebar(Notsos)
  Dtrprogram( DFHDYP    )       Sosaboveline(Notsos)
  Dumping( Sysdump )            Sosbelowline(Notsos)
  Edsalimit( 0268435456 )       Storeprotect(Active)
  Forceqr( Noforce )            Time( 0001000 )
  Logdefer( 00030 )             Tranisolate(Active)
  Maxtasks( 075 )
  Memlimit(Nolimit)
  MQconn ( )
  Mrobatch( 001 )
  Oslevel(010400)


                                              SYSID=HT63 APPLID=CICSHT63
  RESPONSE: NORMAL                          TIME:  16.00.11  DATE: 10.03.01
PF 1 HELP       3 END       5 VAR       7 SBH 8 SFH 9 MSG 10 SB 11 SF
```

Figure 101. CEMT INQUIRE SYSTEM screen

CEMT INQUIRE SYSTEM

Displayed fields

AGING (*value***)**
Displays the time factor for CICS to use in the priority aging algorithm for incrementing the priority of a task. The value is expressed as "milliseconds per unit of priority", and is in the range 0–65535.

The priority aging factor is defined initially by the **PRTYAGE** system initialization parameter, in which the value represents the number of milliseconds that must elapse before the priority of a waiting task can be adjusted upward by 1. See the *CICS System Definition Guide* for more information.

Note: You can reset this value by overtyping it with a different value.

AKP (*value*)
Displays the activity keypoint trigger value, which is the number of logging operations between the taking of keypoints. The value is either zero, or in the range 200–65535. A value of zero indicates that there is no activity keypointing.

Note: You can reset this value by overtyping it with a different value.

CICSTSLEVEL (*value*)
Displays a 6-character value identifying the version, release, and modification level of the CICS Transaction Server for z/OS product under which the CICS region is running. The value is of the form *vvrrmm*. For example, CICS Transaction Server Version 4 Release 2 returns 040200.

CMDPROTECT (*value*)
Displays whether command protection, which validates start addresses passed on in CICS commands, is active or not (that is, whether the **CMDPROT** system initialization parameter specifies YES or NO). The values are as follows:

CMDPROT
Command protection is active. CICS checks to ensure that the task itself has write access to the storage referenced on the command before writing to the storage on behalf of the task.

NOCMDPROT
Command protection is not active. CICS does not check to ensure that the task itself has write access to the storage referenced on the command before writing to the storage on behalf of the task.

DB2CONN (*value*)
Displays the name of the currently installed DB2CONN resource definition. Only one DB2CONN definition can be installed at a time. The DB2CONN resource definition specifies the attributes of the connection between CICS and DB2.

DEBUGTOOL(*value*)
Displays whether you can use debugging profiles to select the programs that will run under the control of a debugging tool. The following debugging tools use debugging profiles:

- Debug Tool, for compiled language application programs (programs written in COBOL, PL/I, C, C++, and Assembler)
- Remote debugging tools (for compiled language application programs and Java programs)

Other debugging mechanisms, such as the CICS Execution Diagnostic Facility (CEDF) do not use debugging profiles.

The values are as follows:

DEBUG
You can use CICS debugging profiles to select the programs that will run under the control of a debugging tool.

NODEBUG
You cannot use CICS debugging profiles to select the programs that will run under the control of a debugging tool.

For more information about debugging profiles, see the *CICS Application Programming Guide*.

DFLTUSER (*value*)
Displays the user ID of the default user for this instance of the CICS region.

DSALIMIT (*value*)
Displays the maximum amount of storage, as a total number of bytes, within which CICS can dynamically allocate storage for the four individual DSAs that reside below 16 MB (below the line). For more information, see DSALIM system initialization parameter in the System Definition Guide.

Note: You can reset this value by overtyping it with a different value.

If DSALIMIT specifies a value lower than the current limit, CICS might not implement the new limit immediately, but attempts to do so over time as dynamic storage is freed in the individual DSAs.

DSRTPROGRAM (*pgrmid*)
Displays the name of the distributed routing program.

Note: You can reset this value by overtyping it with a different value.

DTRPROGRAM (*pgrmid*)
Displays the name of the dynamic routing program.

Note: You can reset this value by overtyping it with a different value.

DUMPING(*value*)
Displays whether CICS system dumps are suppressed. The values are as follows:
NOSYSDUMP
System dumps are suppressed.
SYSDUMP
System dumps are not suppressed.

These values are set by the SIT parameter DUMP=YES or NO.

EDSALIMIT (*value*)
Displays the maximum amount of storage, as a total number of bytes, within which CICS can dynamically allocate storage for the individual DSAs that reside above 16 MB but below 2 GB (above the line). For more information, see EDSALIM system initialization parameter in the System Definition Guide.

Note: You can reset this value by overtyping it with a different value.

If EDSALIMIT specifies a value lower than the current limit, CICS might not implement the new limit immediately, but attempts to do so over time as dynamic storage is freed in the individual DSAs.

FORCEQR (*value*)
Displays whether CICS is to CICSAPI force user application programs that are defined as threadsafe to run on the quasi-reentrant (QR) TCB. The values are as follows:
FORCE
CICS forces all user application programs specified with the CONCURRENCY(THREADSAFE) attribute to run under the QR TCB, as if they were specified with CONCURRENCY(QUASIRENT). Force does not apply to certain programs, for example OPENAPI programs, or C or C++ programs compiled with XPLINK. For details, see FORCEQR system initialization parameter in the System Definition Guide.
NFORCE
CICS is to honor the CONCURRENCY(THREADSAFE) attribute on CICSAPI user application programs, and allow user programs to run on an open TCB to avoid unnecessary TCB switching.

LOGDEFER (*value***)**
Displays the log deferral interval in milliseconds, which is the period of time used by CICS Log Manager when determining how long to delay a forced journal write request before invoking the MVS system logger. It is in the range 0-65535. See the *CICS System Definition Guide* for information about the **LOGDEFER** parameter and associated system initialization parameter **LGDFINT**.

Note: You can reset this value by overtyping it with a different value.

MAXTASKS (*value***)**
Displays the maximum number of tasks, both active and suspended, allowed at any one time in the CICS system. It must have a value in the range 1–999.

Note:

1. You can reset this value by overtyping it with a different value.
2. The value assigned to MAXTASKS might be less than the requested value, because of CICS storage constraints. If this occurs, the message CEILING REACHED is displayed when the request is made.

MEMLIMIT (*value***)**
Displays the limit of storage above the bar for use by the CICS region. A value of NOLIMIT indicates that no limit is imposed on the amount of storage that the region can attempt to use.

MQCONN(*value***)**
Displays the name of the MQCONN resource definition that is currently installed for the CICS region. If no MQCONN resource definition is currently installed, the field is blank. Only one MQCONN resource definition can be installed at a time in a CICS region. The MQCONN resource definition specifies the attributes of the connection between CICS and WebSphere MQ.

MROBATCH (*value***)**
Displays the number of MRO requests from connected regions that are to be batched before this region is posted. It is a value in the range 1–255.

Note: You can reset this value by overtyping it with a different value.

OSLEVEL (*value***)**
Displays the version, release, and modification level of z/OS on which CICS is running. For example, z/OS Version 1 Release 4 Modification 0 appears as 010400.

PROGAUTOCTLG
Displays whether autoinstalled program definitions are to be cataloged. The values are as follows:

CTLGALL
All autoinstalled program definitions are to be cataloged and restored on a warm or emergency start.

CTLGMODIFY
Autoinstalled program definitions are to be cataloged only if they are modified (for example, by a CEMT SET PROGRAM command), so that the modified definitions are restored on a warm or emergency restart.

CTLGNONE
No autoinstalled program definitions are to be cataloged. They are autoinstalled again after a warm or emergency start.

Note: You can reset this value by overtyping it with a different value.

PROGAUTOEXIT (*value*)
Displays the name of the user-provided program that is called by the program autoinstall code to select or modify a model definition.

Note: You can reset this value by overtyping it with a different value.

PROGAUTOINST
Displays whether autoinstall for programs is active or inactive. The values are as follows:

AUTOACTIVE
Autoinstall for programs is active. On first use, if a program, map set, or partition set is not defined, the definition is created dynamically.

AUTOINACTIVE
Autoinstall is not active. If a program is not defined, a PGMIDERR or transaction abend occurs when it is referenced.

Note: You can reset this value by overtyping it with a different value.

REENTPROTECT (*value*)
Displays whether read-only storage is in use for reentrant programs (that is, whether the RENTPGM system initialization parameter specifies PROTECT or NOPROTECT). The values are as follows:

NOREENTPROT
CICS allocates storage for the read-only DSAs (RDSA and ERDSA) from CICS-key storage. Reentrant programs do not have the protection of residing in read-only storage, and can be modified by programs executing in CICS key.

REENTPROT
CICS allocates storage for the RDSA and ERDSA from key-0, non-fetch protected, storage. CICS loads reentrant programs into this storage, and the programs are protected by residing in read-only storage.

RELEASE (*value*) (SUPPORTED FOR COMPATIBILITY ONLY)
Displays a 4-digit number (value 0670 for this release of CICS TS) representing the level of the CICS code running this region.

This field is maintained for compatibility with previous releases of CICS only. As an exclusive element of CICS Transaction Server for z/OS, CICS does not have a product version and release number of its own. To determine the release number of the product, see the value displayed for CICSTSLEVEL.

RUNAWAY (*value*)
Displays the interval, in milliseconds, for which a task can have control before it is assumed to be looping.

Note: You can reset this value by overtyping it with a different value.

If you are resetting this value, you can specify 0, or a value in the range 500–2700000. The value you specify is rounded down to a multiple of 500.

CICS purges a task if the task has not given up control after this interval (that is, CICS assumes that the task is looping). If you specify zero, runaway task control is inoperative (that is, tasks do not get purged if they appear to be looping). The value is independent of, and can be less than, the value in the TIME operand.

CICS runaway-task detection is based upon task time (that is, the interval is reset each time a task receives control of the processor, and is then decremented while the task is in control). You do not, therefore, need to allow for the effect of other jobs when setting the runaway value.

SCANDELAY (*value*)
Displays the maximum number of milliseconds between the receipt of a z/OS Communications Server request, and the time when CICS begins to process the request. The number is in the range 0–5000, and is less than or equal to the TIME value.

Note:

1. You can reset this value by overtyping it with a different value.
2. In the summary report of the dispatcher statistics, SCANDELAY is referred to as ICVTSD.

SDTRAN (*value*)
Displays the 4-character name of the transaction to be run at the beginning of normal or immediate shutdown. This may be the name of a user-supplied transaction, or the CICS-supplied default transaction, CESD.

SOSABOVEBAR(*value*)
Displays whether CICS is short on storage in the dynamic storage areas above the bar.

NOTSOS
CICS is not short on storage in any of the dynamic storage areas above the bar.

SOS CICS is short on storage in at least one of the dynamic storage areas above the bar.

SOSABOVELINE(*value*)
Displays whether CICS is short on storage in the dynamic storage areas above 16 MB but below 2 GB (above the line).

NOTSOS
CICS is not short on storage in any of the dynamic storage areas above 16 MB but below 2 GB.

SOS CICS is short on storage in at least one of the dynamic storage areas above 16 MB but below 2 GB.

SOSBELOWLINE(*value*)
Displays whether CICS is short on storage in the dynamic storage areas below 16 MB (below the line).

NOTSOS
CICS is not short on storage in any of the dynamic storage areas below 16 MB.

SOS CICS is short on storage in at least one of the dynamic storage areas below 16 MB.

STOREPROTECT (*value*)
Displays whether storage protection is active in the CICS region. The values returned are:

ACTIVE
CICS is operating with the CICS storage protection facility.

INACTIVE
CICS is operating without the CICS storage protection facility.

See STGPROT system initialization parameter in the System Definition Guide for more information about CICS storage protection.

TIME (*value*)
Displays the interval, in milliseconds, for which CICS releases control to the operating system if no transactions are ready to resume processing. This interval is known as the *region exit interval*.

Note:

1. You can reset this value by overtyping it with a different value. It can have a value in the range 100–3600000, and must be greater than or equal to the SCANDELAY value.
2. In the summary report of the dispatcher statistics, TIME is referred to as ICV.

TRANISOLATE
Displays the status of transaction isolation. The values returned are:
ACTIVE
Transaction isolation is active in the CICS region.
INACTIVE
CICS is running without transaction isolation, either because the support is not available, or it was not requested at CICS initialization.

CEMT INQUIRE TASK

Retrieve information about a user task.

In the CICS Explorer, the Tasks operations view provides a functional equivalent to this command.

Description

INQUIRE TASK returns information about user tasks. Only information about user tasks can be displayed or changed; information about CICS-generated system tasks or subtasks cannot be displayed or changed. System tasks are those tasks started and used internally by CICS, and not as a result of a user transaction.

Input

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT INQUIRE TASK` (or suitable abbreviations for the keywords). The resulting display lists the status of every task.
- Type `CEMT INQUIRE TASK` (or suitable abbreviations for the keywords), followed by the attributes that are necessary to limit the range of information that you require. For example, if you enter `CEMT INQ TA TE`, the resulting display shows the details of only those tasks that were started from a terminal.

You can change various attributes in the following ways:

- Overtype your changes on the INQUIRE screen after tabbing to the appropriate field (see "Overtyping a display" on page 117).
- Use the **CEMT SET TASK** command.

(*value***)**
The CICS-generated task number, in the range 1 - 99999.

ALL
The default value. The maximum number of tasks displayed is 32000.

TCLASS (*value***)**
The 8-character transaction class name to which the transaction belongs. The maximum number of tasks displayed is 32000.

You cannot specify a list of identifiers, or use the asterisk (*) or plus (+) symbols to specify a family of tasks.

When a value does not apply, or is negative (the value begins with No), the fields on the screen are blank. To modify these fields, locate them by tabbing (they appear in the same sequence as in the expanded format), and overtype with input valid for that field. You might find it more convenient to use the expanded format when setting one of these values.

If you place the cursor against a specific entry in the list and press ENTER, CICS displays an expanded format.

CEMT INQUIRE TASK

Displayed fields

ACTIVITY *(value)*
Displays the 16-character, user-assigned, name of the CICS business transaction services activity that this task is executing on behalf of.

ACTIVITYID *(value)*
Displays the 52-character, CICS-assigned, identifier of the CICS business transaction services activity that this task is executing on behalf of.

BRFACILITY(*value***)**
Displays the 8-byte facility token representing the virtual terminal used by the current task if it is used by the 3270 bridge mechanism. If the task is not currently running in the 3270 bridge environment, zeros are returned.

BRIDGE (*value***)**
Displays the 4-character name of the bridge monitor transaction if the current task is running in a 3270 bridge environment, and was started by a bridge monitor transaction with a **START BREXIT TRANSID** command. Otherwise, blanks are returned.

DB2PLAN (*value***)**
Displays the 1- to 8-character name of the DB2 plan being used by this task, or blanks if no DB2 plan is being used.

FACILITY (*value***)**
Displays a 4-character string identifying the name of the terminal or queue that initiated the task. If no FACILITY value is displayed, the task was started without a facility.

FTYPE
Displays the type of facility that initiated this task. The values are as follows:
TASK
The task was initiated from another task.
TERM
The task was initiated from a terminal.
DEST
The task was initiated by a destination trigger level as defined in the TDQUEUE resource definition.

HTIME (*value***)**
Displays the time (in seconds) that the task has been in the current suspended state.

HTYPE (*value***)**
Displays the reason why the task is suspended. A null value indicates that there is no hold-up, except for the necessity of reaching the head of the queue.

HVALUE (*value***)**
Displays a 16-character resource name, such as a file name, or a value such as a TCLASS value.

For information about the values that CICS can return in the HTYPE and HVALUE options, see the resource type and resource name details in the *CICS Problem Determination Guide*.

INDOUBTMINS (*value***)**
Displays the length of time, in minutes, after a failure during the indoubt period, before the task is to take the action returned in the INDOUBT attribute. The returned value is valid only if the unit of work is indoubt and the value of the INDOUBTWAIT attribute returns WAIT.

INDOUBT (*value***)**
Displays the action (based on the ACTION attribute of the TRANSACTION resource definition) to be taken if the CICS region fails or loses connectivity with its coordinator while a unit of work is in the indoubt period.

The action is dependent on the values returned in the INDOUBTWAIT and INDOUBTMINS attributes; if INDOUBTWAIT returns WAIT, the action is not taken until the time returned in INDOUBTWAIT expires.

The values are as follows:

BACKOUT
All changes made to recoverable resources are to be backed out.

COMMIT
All changes made to recoverable resources are to be committed, and the unit of work marked as completed.

INDOUBTWAIT (*value***)**
Displays how (based on the WAIT attribute of the TRANSACTION definition) a unit of work (UOW) is to respond if a failure occurs while it is in an indoubt state. The values are as follows:

NOWAIT
The UOW is not to wait, pending recovery from the failure. CICS is to take immediately whatever action is specified on the ACTION attribute of the TRANSACTION definition.

WAIT
The UOW is to wait, pending recovery from the failure, to determine whether recoverable resources are to be backed out or committed.

For further information about the meaning of the ACTION and WAIT attributes of the TRANSACTION definition, see TRANSACTION definition attributes.

IDENTIFIER (*value***)**
Displays a 48-character field that contains user data provided by the bridge exit, if the task was initiated in the 3270 bridge environment, or blanks, otherwise. This field is intended to assist in online problem resolution.

For example, this field might contain the WebSphere MQ correlator for the CICS-MQ bridge, or a web token.

PRIORITY (*value***)**
Displays the priority of the task in the range 0 - 255, where 255 is the highest priority.

Note: You can reset this value by typing over it with a different value.

PROCESS *(value)*
Displays the 36-character name of the CICS business transaction services process that this task is executing on behalf of.

PROCESSTYPE *(value)*
Displays the 8-character process-type of the CICS business transaction services process that this task is executing on behalf of.

PURGETYPE (*input only field***)**
Specifies whether a task is to be purged or forced to purge. The values are as follows:

PURGE
The task is to be terminated. Termination occurs only when system and data integrity can be maintained.

FORCEPURGE
The task is to be terminated immediately. System integrity is not guaranteed. In some extreme cases, for example if a task is forced to

purge during backout processing, CICS terminates abnormally. If you want to terminate a task but do not want to terminate CICS, use PURGE instead of FORCEPURGE.

RUNSTATUS
Displays the status of this task. The values are as follows:

RUNNING
The task is running.

DISPATCHABLE
The task is dispatchable.

SUSPENDED
The task is suspended.

STARTCODE (*value*)
Displays how this task was started. The values are as follows:

D A distributed program link (DPL) request. The program cannot issue I/O requests against its principal facility or any sync point requests.

DS A distributed program link (DPL) request, as for code D, with the exception that the program can issue sync point requests.

QD A transient data trigger level was reached.

S Start command (no data)

SD Start command (with data)

TO The operator typed a transaction code at the terminal.

TP The transaction was started by presetting the transaction ID for the terminal.

U User-attached task.

Note: When the IIOP request processor is run locally the start code for an **ASSIGN** command or an **INQUIRE TASK** is U. When the IIOP request processor is run remotely, over an MRO link, the start code for these commands is T0. If you attempt to run the IIOP request processor remotely over any other type of connection, the routing request is not accepted. Start codes for these commands are not relevant in this situation.

TASK (*value*)
Indicates that this panel relates to a TASK inquiry and displays a CICS-generated task number in the range 1–99999.

TCB (*value*)
Displays the type of TCB under which the task is running. The values are as follows:

CKOPEN
The task is running under a CICS key open TCB.

INTERNAL
The task is running under one of the CICS internal TCBs. An internal TCB can be one of the following:

- The concurrent mode (CO) TCB
- The file-owning mode (FO) TCB
- The shared class cache (JM) TCB
- The resource-owning mode (RO) TCB
- The ONC/RPC mode (RP) TCB
- The sockets listener mode (SL) TCB
- The secure sockets layer mode (SO) TCB
- A sockets mode (S8) TCB
- The FEPI mode (SZ) TCB.

QR The task is running under the CICS QR TCB.

UKOPEN
The task is running under a user key open TCB.

TRANID (*value*)
Displays a 4-character string identifying the transaction name associated with the task.

UOW (*value*)
Displays the 16-character local identifier of the unit of work that is associated with this task.

USERID (*value*)
Displays the user that is currently associated with the task.

CEMT INQUIRE TCLASS

Retrieve information about tasks in a task class.

In the CICS Explorer, the Transaction classes operations view provides a functional equivalent to this command.

Description

The **INQUIRE TCLASS** command returns information about the current and maximum number of tasks, the purge threshold, and the status of a user-defined task class.

The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT INQUIRE TCLASS` (or suitable abbreviations for the keywords). The resulting display lists the current status.
- Type `CEMT INQUIRE TCLASS` (or suitable abbreviations for the keywords), followed by the attributes that are necessary to limit the range of information that you require. For example, if you enter `cemt i tc p(value)`, the resulting display shows the details of only those transaction classes that have a purge threshold of the value indicated.

To change various attributes, use the CEMT SET TCLASS command.

(*value*)
The 8-character transaction class name.

ALL
The default.

Sample screen

```
 IN TCLASS
 STATUS:  RESULTS - OVERTYPE TO MODIFY
  Tcl(DFHTCIND) Max( 010 ) Act(000) Pur( 0000000 )
     Que(000000)
  Tcl(DFHTCL01) Max( 001 ) Act(000) Pur( 0000000 )
     Que(000000)
  Tcl(DFHTCL02) Max( 001 ) Act(000) Pur( 0000000 )
     Que(000000)
  Tcl(DFHTCL03) Max( 001 ) Act(000) Pur( 0000000 )
     Que(000000)
  Tcl(DFHTCL04) Max( 001 ) Act(000) Pur( 0000000 )
     Que(000000)
  Tcl(DFHTCL05) Max( 001 ) Act(000) Pur( 0000000 )
     Que(000000)
  Tcl(DFHTCL06) Max( 001 ) Act(000) Pur( 0000000 )
     Que(000000)
  Tcl(DFHTCL07) Max( 001 ) Act(000) Pur( 0000000 )
     Que(000000)
+ Tcl(DFHTCL08) Max( 001 ) Act(000) Pur( 0000000 )
     Que(000000)
```

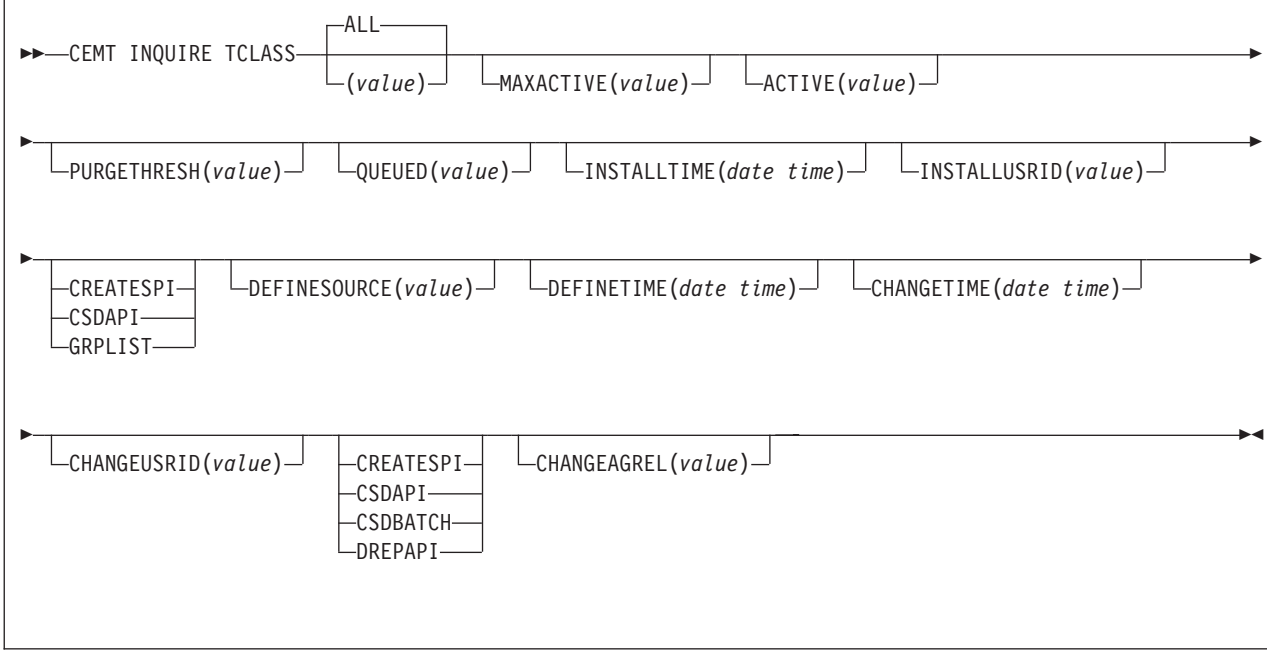
Figure 102. CEMT INQUIRE TASK screen

If you place the cursor against a specific entry in the list and press ENTER, CICS displays an expanded format, as shown in Figure 103.

```
 IN TCLASS
 RESULT - OVERTYPE TO MODIFY
   Tclass(DFHTCIND)
   Maxactive( 010 )
```

Figure 103. The expanded display of an individual entry

CEMT INQUIRE TCLASS



Displayed fields

ACTIVE (*value***)**
Displays the total number of transactions that are currently active in a user-defined transaction class.

CHANGEAGENT(*value***)**
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

DEFINESOURCE(*value***)**
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

MAXACTIVE (*value***)**
Displays the largest number of transactions in the transaction class that are allowed to run concurrently. The value can be in the range 0 - 999.

You can reset this value by overtyping it with a different value.

PURGETHRESH (*value***)**
Displays the limit at which the queuing transactions are purged for the transaction class. It can have a value in the range 0000000 - 1000000. Zero means that the transactions are not purged. A nonzero number "n" means that the first (n-1) transactions to arrive are queued and the nth is purged.

You can reset this value by overtyping it with a different value.

QUEUED (*value***)**
Displays the total number of current tasks that are suspended because the class maximum has been reached.

TCLASS (*value***)**
Indicates that this panel relates to a TCLASS inquiry and displays an 8-character transaction class name.

CEMT INQUIRE TCPIP

Inquire about CICS internal TCP/IP support status.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

The **INQUIRE TCPIP** command returns information about the current status of CICS internal TCP/IP support.

Input

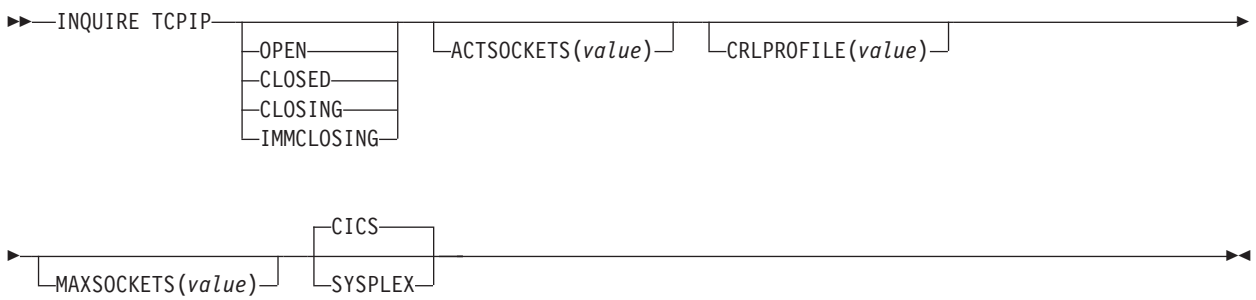
Press the Clear key to clear the screen. There are two ways to start this transaction:

- Type `CEMT INQUIRE TCPIP` (or suitable abbreviations for the keywords). The resulting display lists the current status.
- Type `CEMT INQUIRE TCPIP` (or suitable abbreviations for the keywords), followed by the other attribute settings that you want to view.

You can change various attributes as follows:

- Overtype your changes on the INQUIRE screen after tabbing to the appropriate field (see "Overtyping a display" on page 117).
- Use the **CEMT SET TCPIP** command.

CEMT INQUIRE TCPIP



Displayed fields

ACTSOCKETS(*value*)
Displays the current number of active IP sockets managed by the CICS sockets domain.

CRLPROFILE(*value*)
Displays the name of the profile that authorizes CICS to access an LDAP server that stores certificate revocation lists for SSL connections.

MAXSOCKETS (*value*)
Displays the maximum number of IP sockets that can be managed by the CICS sockets domain.

OPENSTATUS
Displays the status of CICS internal TCP/IP support. The values are:

OPEN CICS internal sockets support is open.

CLOSED
CICS internal sockets support has not yet been activated, or has been terminated.

CLOSING
CICS internal sockets support is in the process of closing.

IMMCLOSING
CICS internal sockets support is in the process of immediate termination.

SSLCACHE(*value*)
Displays whether CICS is configured to use local or sysplex caching of session ids. The values are:

CICS CICS is configured to cache session ids in the local CICS region.

SYSPLEX
CICS is configured to cache session ids in the coupling facility.

CEMT INQUIRE TCPIPSERVICE

Retrieve information about TCP/IP ports on which CICS internal TCP/IP support is currently listening on behalf of other CICS services.

In the CICS Explorer, the TCP/IP services operations view provides a functional equivalent to this command.

Description

The **INQUIRE TCPIPSERVICE** command returns information about the state of a service by using CICS internal TCP/IP support.

The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

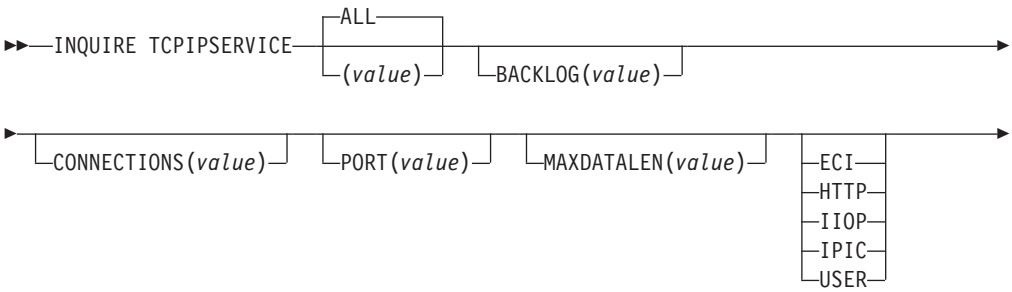
Press the Clear key to clear the screen. You can start this transaction in two ways:

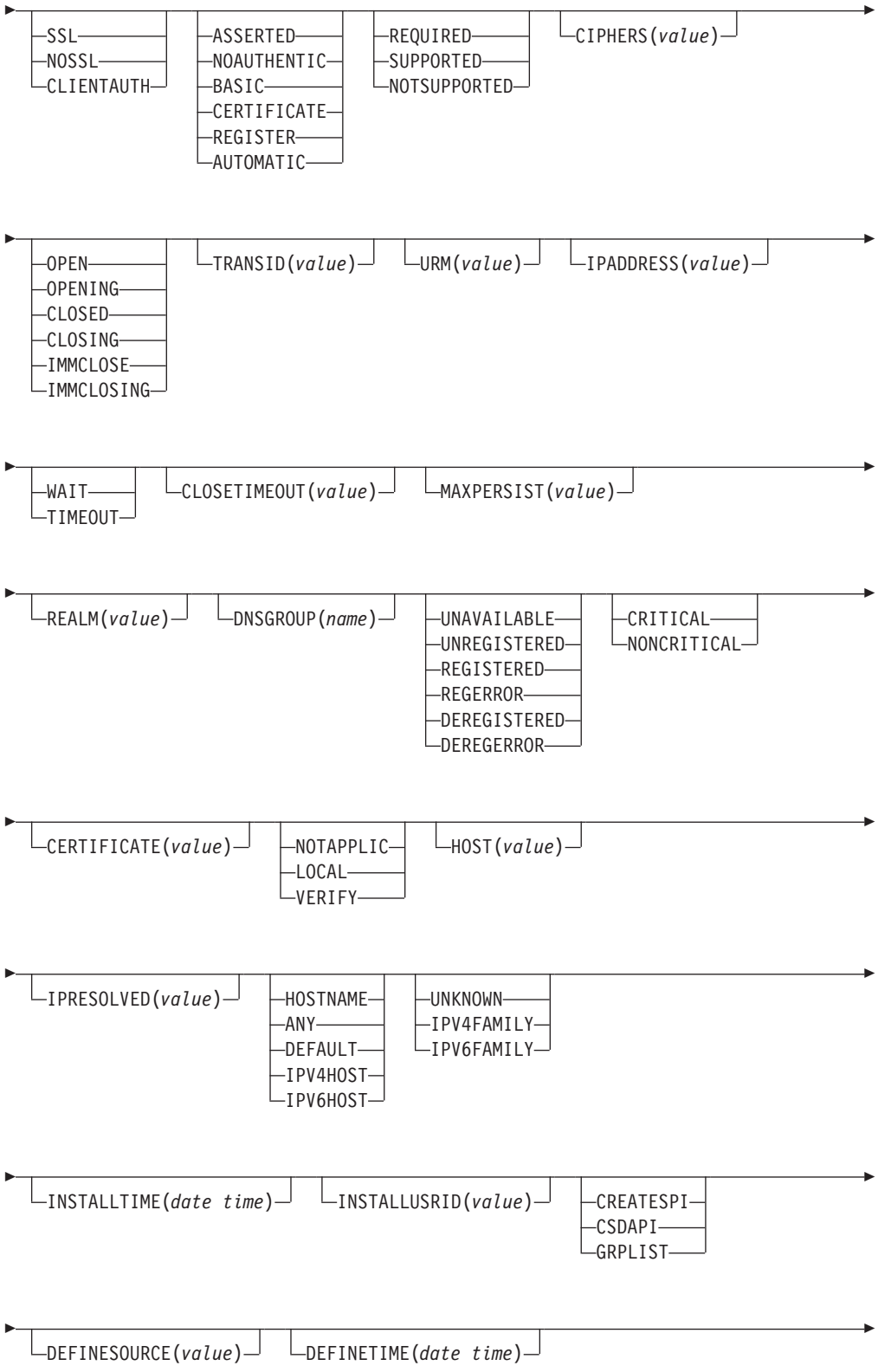
- Type `CEMT INQUIRE TCPIPSERVICE` (or suitable abbreviations for the keywords). The resulting display lists the status.
- Type `CEMT INQUIRE TCPIPSERVICE` (or suitable abbreviations for the keywords), followed by as many of the other attribute settings that you want to view.

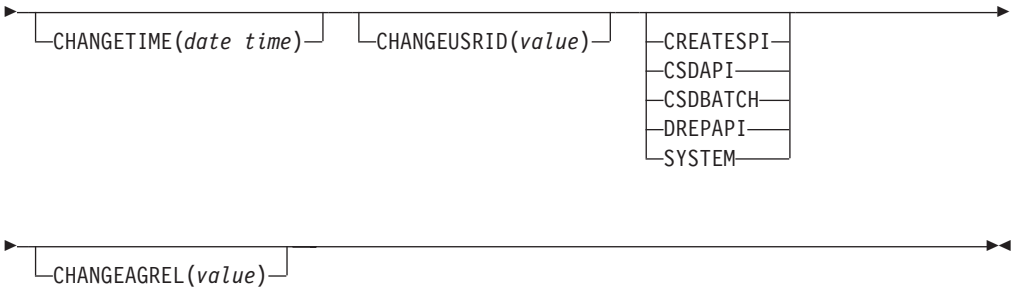
You can change various attributes in the following ways:

- Overtype your changes on the INQUIRE screen after tabbing to the appropriate field.
- Use the CEMT SET TCPIPSERVICE command.

CEMT INQUIRE TCPIPSERVICE




SSL
NOSSL
CLIENTAUTH
ASSERTED
NOAUTHENTIC
BASIC
CERTIFICATE
REGISTER
AUTOMATIC
REQUIRED
SUPPORTED
NOTSUPPORTED
CIPHERS(value)
OPEN
OPENING
CLOSED
CLOSING
IMMCLOSE
IMMCLOSING
TRANSID(value)
URM(value)
IPADDRESS(value)
WAIT
TIMEOUT
CLOSETIMEOUT(value)
MAXPERSIST(value)
REALM(value)
DNSGROUP(name)
UNAVAILABLE
UNREGISTERED
REGISTERED
REGERROR
DEREGISTERED
DEREGERROR
CRITICAL
NONCRITICAL
CERTIFICATE(value)
NOTAPPLIC
LOCAL
VERIFY
HOST(value)
IPRESOLVED(value)
HOSTNAME
ANY
DEFAULT
IPV4HOST
IPV6HOST
UNKNOWN
IPV4FAMILY
IPV6FAMILY
INSTALLTIME(date time)
INSTALLUSRID(value)
CREATESPI
CSDAPI
GRPLIST
DEFINESOURCE(value)
DEFINETIME(date time)



Displayed fields

ATTACHSEC(*value*)
Indicates the level of attach-time security used by the connection.

LOCAL
CICS does not require a user ID or password from clients.

NOTAPPLIC
This option has no meaning for web interface or IIOP TCP/IP services.

VERIFY
Incoming attach requests must specify a user ID and a user password.

AUTHENTICATE(*value*)
Indicates the scheme used to authenticate clients.

ASSERTED
Asserted identity authentication is used. This value is returned only when PROTOCOL has a value of IIOP.

BASIC
HTTP basic authentication is used to obtain a user ID and password from the client. This value is returned only when PROTOCOL has a value of HTTP.

CERTIFICATE
SSL client certificate authentication is used to authenticate and identify the client. This value is returned only when PROTOCOL has a value of HTTP or IIOP.

NOAUTHENTIC
The client is not required to send authentication or identification information. However, if the client sends a valid certificate that is already registered to the security manager, and associated with a user ID, that user ID identifies the client. This value is returned only when PROTOCOL has a value of HTTP or IIOP.

REGISTER
SSL client certificate authentication is used to authenticate the client. If the client certificate is not associated with a user ID, HTTP basic authentication is used to obtain the client user ID, and associate it with the certificate. This value is returned only when PROTOCOL has a value of HTTP.

AUTOMATIC
If the client does not send a certificate, HTTP basic authentication is used to obtain a user ID and password from the client. Otherwise, SSL client certificate authentication is used to authenticate the client. If the

client certificate is not associated with a user ID, HTTP basic authentication is used to obtain the client user ID, and associate it with the certificate. This value is returned only when PROTOCOL has a value of HTTP.

For more information about authentication and identification of HTTP and IIOP clients, see the *CICS RACF Security Guide*.

BACKLOG (*value***)**
Change the maximum number of requests that can be queued in TCP/IP waiting to be processed by the service.

CERTIFICATE (*value***)**
Specifies the label of an X.509 certificate that is to be used in the SSL handshake for the TCP/IP service. Certificate labels can be up to 32 bytes, and are specified for the CICS region user ID in a key ring defined in the database of the external security manager; for example, as defined by the RACDCERT command. If this attribute is omitted, the default certificate defined in the key ring for the CICS region user ID is used.

CHANGEAGENT(*value***)**
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

SYSTEM
The resource definition was last changed by the CICS or CICSPlex system.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

CIPHERS(*value***)**
Returns a 56-character area containing the list of cipher suites that is used to negotiate with clients during the SSL handshake. The list is set by the ENCRYPTION system initialization parameter, but you can edit the list to remove or change the order of cipher suites.

CLOSETIMEOUT (*value*)
Returns a fullword value containing the number of seconds specified for the timeout period. This value can be in the range 0 - 86400 (24 hours). For the HTTP protocol, do not specify 0 because this setting means that persistent connections cannot be maintained.

CONNECTION (*value*)
The number of current sockets connections for this service

DEFINESOURCE(*value*)
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time*)
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

DNSGROUP(*name*)
Returns the 18-character Domain Name System (DNS) group name that CICS registers with the z/OS Workload Manager (WLM) for this TCPIPSERVICE resource.

DNSSTATUS(*value*)
Returns the current state of WLM/DNS registration of this TCPIPSERVICE resource.

UNAVAILABLE
Registration is not supported by z/OS.

UNREGISTERED
Registration has not yet occurred. This state is the initial state of any service.

REGISTERED
Registration completed successfully.

REGERROR
Registration failed with an error.

DEREGISTERED
Deregistration completed successfully.

DEREGERROR
Deregistration failed with an error.

GRPCRITICAL(*value*)
Indicates whether this TCPIPSERVICE resource is a critical member of the DNS group.

CRITICAL
If this TCPIPSERVICE resource is closed, or stops listening for an unusual reason, the group name specified in the DNSGROUP attribute is deregistered from WLM.

NONCRITICAL
If this TCPIPSERVICE resource is closed, or stops listening for an unusual reason, the group name specified in the DNSGROUP attribute is not deregistered from WLM, unless this service is the last one in a set with the same group name.

HOST(*value*)
Displays the host name, IPv4, or IPv6 address of the remote system. All IPv4 addresses are displayed as native IPv4 dotted decimal addresses, for example, `1.2.3.4`, regardless of the type of address format used. For information about accepted IPv4 formats, see the *CICS Internet Guide*.

HOSTTYPE(*value*)
Displays the address format of the HOST option. The HOSTTYPE option is available using an expanded query only. The possible values are as follows:

HOSTNAME
The HOST option contains a character host name.

ANY The ANY option is specified for the HOST option.

DEFAULT
The DEFAULT option is specified for the HOST option.

IPV4HOST
The address is an IPv4 address.

IPV6HOST
The address is an IPv6 address.

INSTALLAGENT(*value*)
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

INSTALLTIME(*date time*)
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value*)
Displays the 8-character user ID that installed the resource.

IPADDRESS (*value*)
Returns an IP address. If you specify an IP address in the TCPIPSERVICE definition, that address is returned; otherwise, the default IP address is returned. If a z/OS system has more than one IP address only the default is returned.

IPFAMILY(*value*)
Displays the address format of the IPRESOLVED option. The IPFAMILY option is available by using an expanded query only. Filtering interacts with the IPFAMILY option when you filter by using wildcard characters. For example, if the value in IPFAMILY is IPV6FAMILY, you must use the colon character to retrieve an IP address. The possible values are as follows:

IPV4FAMILY
The address is an IPv4 address.

IPV6FAMILY
The address is an IPv6 address.

UNKNOWN
The IPRESOLVED option is not yet in use, or the address cannot be resolved.

IPRESOLVED(*value***)**
Displays the IPv4 or IPv6 address of the host. The IPRESOLVED option is available by using an expanded query only.

MAXDATALEN (*value***)**
Returns a fullword value containing the maximum length of data that can be received by CICS as an HTTP server. This value can be 3 KB through 524288 KB. The default is 32 KB.

MAXPERSIST(*value***)**
Returns a fullword value containing the maximum number of persistent connections from web clients that the CICS region allows for this port at any one time. This setting applies only for the HTTP protocol. NO means that there is no limit on the number of persistent connections. A zero setting means that no persistent connections are allowed. A zero setting is not compliant with the HTTP/1.1 specification and must not be set in a CICS region that is handling external requests.

OPENSTATUS(*value***)**
Returns the state of the service.

OPEN CICS internal sockets support is open.

OPENING
CICS internal sockets support is in the process of opening.

CLOSED
CICS internal sockets support is not yet activated, or has been terminated.

CLOSING
CICS internal sockets support is in the process of closing.

IMMCLOSE
CICS internal sockets support has immediately terminated for this service.

IMMCLOSING
CICS internal sockets support is in the process of immediate termination.

PORT (*value***)**
Returns the number of the port on which CICS is listening on behalf of this service.

PRIVACY(*value***)**
Returns the level of SSL encryption required for inbound connections to this service.

REQUIRED
Encryption must be used. During the SSL handshake, CICS advertises only supported cipher suites that provide encryption.

SUPPORTED
Encryption is used if both client and server support it. During the SSL handshake, CICS advertises all supported cipher suites.

NOTSUPPORTED
Encryption must not be used. During the SSL handshake, CICS advertises only supported cipher suites that do not provide encryption.

PROTOCOL(*value***)**
Identifies to CICS the type of service to be provided on the TCP/IP port.

ECI Connections are handled by CICS ECI over TCP/IP support.

HTTP
Connections are handled by CICS web support.

IIOP Connections are handled by CICS IIOP support. IIOP is required for TCPIPSERVICEs that are to accept inbound requests for enterprise beans.

IPIC IP interconnectivity.

USER
The user-defined protocol is used. Requests are passed to the analyzer program for the TCPIPSERVICE, and handled by using CICS web support facilities, but the HTTP specifications are not used to check the messages.

REALM (*value***)**
Returns the 56-character realm that is used during the process of HTTP basic authentication.

SOCKETCLOSE(*value***)**
Indicates whether a timeout value is in effect for the TCPIPSERVICE.

WAIT
NO was specified on the definition. Socket receives wait for data indefinitely.

TIMEOUT
A value was specified for the SOCKETCLOSE parameter on the definition.

SSLTYPE(*value***)**
Indicates the status of SSL support for this service.

SSL Nonclient authentication SSL flows are processed on this port.

NOSSL
Non-SSL flows are processed on this port.

CLIENTAUTH
SSL client authentication flows are processed on this port.

TCPIPSERVICE (*value***)**
8-character name identifying this service.

TRANSID (*value***)**
4-character transaction ID used on the attach for the task started to process a new request.

URM (*value***)**
The 8-character name of the user-replaceable program to be started by the attached task.

CEMT INQUIRE TDQUEUE

Retrieve information about transient data queues.

In the CICS Explorer, the TD Queues operations view provides a functional equivalent to this command.

Description

The **INQUIRE TDQUEUE** command returns information about a named transient data (TD) queue that is defined in a TDQUEUE resource definition.

The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT INQUIRE TDQUEUE` (or suitable abbreviations for the keywords). The resulting display lists the current status.
- Type `CEMT INQUIRE TDQUEUE` (or suitable abbreviations for the keywords) followed by the attributes that are necessary to limit the range of information that you require. For example, if you enter `cemt i td ind en`, the resulting display shows the details of only those transient data queues that are indirect and enabled.

You can change various attributes in the following ways:

- Overtype your changes on the INQUIRE screen after tabbing to the appropriate field. See “Overtyping a display” on page 117.
- Use the **CEMT SET TDQUEUE** command.

(*value***)**
The 1 - 4 character identifier of a transient data queue.

ALL
The default.

Sample screen

```
 IN TDQ
 STATUS:  RESULTS - OVERTYPE TO MODIFY
  Tdq(CADL)              Ind Nam(CSSL)
  Tdq(CAIL)              Ind Nam(CSSL)
  Tdq(CCPI)              Ind Nam(CSSL)
  Tdq(CCSE)              Ind Nam(CCSO)
  Tdq(CCSO)              Ext           Ena Ope
                             Mod Out Dat(001) Ddn(COUT    )
  Tdq(CDBC)              Ind Nam(CSSL)
  Tdq(CDUL)              Ind Nam(CSSL)
  Tdq(CESE)              Ext           Ena Ope
                             Shr Out Dat(001) Ddn(CEEMSG  )
  Tdq(CESO)              Ext           Ena Ope
                             Shr Out Dat(001) Ddn(CEEOUT  )
  Tdq(CMIG)              Ind Nam(CSSL)
  Tdq(CRDI)              Ind Nam(CSSL)
+ Tdq(CSCS)              Ind Nam(CSSL)
```

Figure 104. Example CEMT INQUIRE TDQUEUE screen

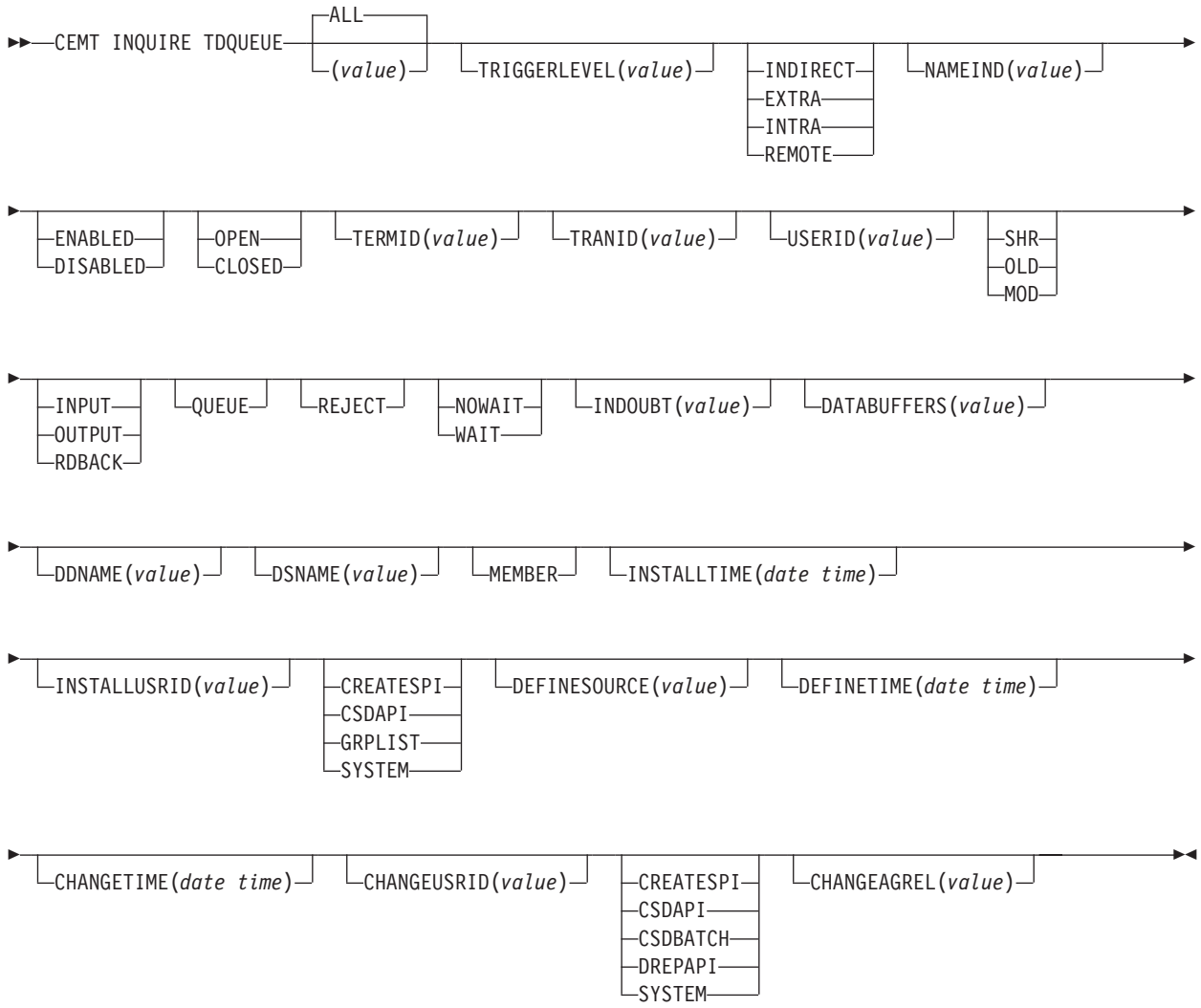
Note: When a value does not apply, or is negative (the value begins with No), the fields on the screen are blank. To modify these fields, locate them by tabbing (they appear in the same sequence as in the expanded format), and overtype with input valid for that field. You might find it more convenient to use the expanded format when setting one of these values.

If you place the cursor against a specific entry in the list and press Enter, CICS displays an expanded format, as shown in Figure 105.

```
IN TDQ
RESULT - OVERTYPE TO MODIFY
  Tdqueue(CCSO)
  Type(Extra)
  Nameind()
  Triggerlevel(       )
  Enablestatus( Enabled )
  Openstatus( Open )
  Termid()
  Tranid()
  Userid()
  Disposition(Mod)
  Iotype(Output)
  Indoubt()
  Indoubtwait()
  Databuffers(001)
  Ddname(COUT)
  Dsname(BELL.CICSHT61.JOB04382.D0000105.?)
```

Figure 105. The expanded display of an individual entry

CEMT INQUIRE TDQUEUE



Displayed fields

CHANGEAGENT(*value*)
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

SYSTEM
The resource definition was last changed by the CICS or CICSPlex system.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

DATABUFFERS (*value) (extrapartition queues only***)**
Displays the number of buffers that can be used by the associated QSAM data set, in the range 1 - 255.

An extrapartition queue can be input or output, but not both.

DDNAME (*value) (extrapartition queues only***)**
Displays a 1- to 8-character value that might refer to a data set defined in the startup JCL.

DEFINESOURCE(*value***)**
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

DISPOSITION (*extrapartition queues only***)**
Displays the status of the associated data set.

SHR The data set existed before this job step and can be read by other concurrent jobs.

OLD The data set existed before this job step.

MOD The system first assumes that the data set exists. For an existing sequential data set, MOD causes the read/write mechanism to be positioned after the last record in the data set. The read/write mechanism is positioned after the last record each time the data set is opened for output.

If the system cannot find volume information for the data set on the DD statement, in the catalog, or passed with the data set from a previous step, the system assumes that the data set is being created in this job step. For a new data set, MOD causes the read/write mechanism to be positioned at the beginning of the data set.

DSNAME (*value***) (extrapartition queues only)**
Displays the name of the associated QSAM data set (1 - 44 characters in

length), or a dummy data set (or dummy data set name) that indicates whether the data set will be used by this transient data queue. This option is applicable for output resources only.

ENABLESTATUS (all except indirect and remote queues)
Displays a value indicating whether the queue can be accessed by applications.

ENABLED
The queue can be accessed by applications.

DISABLED
The queue cannot be accessed by applications, although it can still be open. Queues with names that begin with C cannot be disabled because they are usually reserved for use by CICS.

Note:

1. You can reset this value by overtyping it with a different value.
2. A queue can be disabled only if there are no current users. If units of work (UOWs) are currently using the queue, the queue enters a "disable pending" state. The last UOW to complete its operation on the queue fully disables the queue.
3. When a queue is in a "disable pending" state, the value of the ENABLESTATUS option cannot be altered.

INDOUBT (*value*) (intrapartition queues only)
Indicates the action CICS is to take for an indoubt unit of work (UOW) if the definition for this queue specifies WAIT(YES).

INDOUBTWAIT (intrapartition queues only)
Specifies whether an indoubt unit of work (UOW) (which has modified a logically recoverable queue) should wait for resynchronization with its coordinator to determine whether to commit or back out the changes.

NOWAIT
The UOW is not to wait. Any changes made to recoverable resources are to be backed out or committed, as specified by the ACTION attribute on the transaction resource definition.

WAIT
The UOW is to wait, and any action required while waiting is determined by the WAITACTION attribute.

This attribute overrides the WAIT parameter defined on the UOW transaction definition.

INSTALLAGENT(*value*)
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

SYSTEM
The resource was installed by the CICS or CICSPlex SM system.

INSTALLTIME(*date time*)
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value*)
Displays the 8-character user ID that installed the resource.

IOTYPE (extrapartition queues only)
Displays the type of data set to be used:

INPUT
An input data set.

OUTPUT
An output data set.

RDBACK
An input data set that is to be read backward.

MEMBER (extrapartition queues only)
Returns a 1- to 8-character name that indicates the member name of a partitioned data set. This name is blank if the QSAM data set is not a partitioned data set.

OPENSTATUS (EXTRAPARTITION QUEUES ONLY)
Displays whether the extrapartition queue is open or closed.

OPEN
The queue is open.

CLOSED
The queue is closed.

Note:

1. You can reset this value by overtyping it with a different value.
2. Queues no longer have to be enabled before they can be opened or closed. It is advisable for a queue to be disabled before it is opened or closed.

NAMEIND (*value*) (INDIRECT QUEUES ONLY)
Displays a 4-character string that identifies the name of the queue pointed to by the indirect queue.

QUEUE
The UOW is indoubt and waiting; any locks held by the UOW for this queue remain active until the final state of the UOW is known. Therefore, tasks are suspended rather than receiving the LOCKED response. When the final state of the UOW is known, any changes that it made are committed or backed out. Until then, any further requests of the following types that need one of the active locks must wait:

- READQ, if the indoubt UOW has issued READQ or DELETEQ requests
- WRITEQ, if the indoubt UOW has issued WRITEQ or DELETEQ requests
- DELETEQ, if the indoubt UOW has issued READQ, WRITEQ or DELETEQ requests

REJECT
The UOW is indoubt and is waiting. Any locks held by the UOW for this queue are retained until the final state of the UOW is known. When the final state is known, any changes that the UOW made are committed or backed out. Until then, any further requests that need one of the retained locks are rejected, and a LOCKED response is returned. INDOUBT=REJECT causes a LOCKED

response to be raised in exactly the same circumstances as those in which INDOUBT=QUEUE causes a transaction to wait.

TDQUEUE (*value*)
Indicates that this panel relates to a TDQUEUE inquiry and displays the 4-character identifier of a transient data queue. Queue names that begin with C are normally reserved for use by CICS.

TERMID (*value*)
Displays the 4-character name of the terminal or session to be associated with this queue when automatic transaction initiation (ATI) occurs. See also TRANID and TRIGGERLEVEL.

TRANID (*value*)
Displays the 4-character identifier of the transaction that is to be initiated automatically when the queue trigger level is reached.

TRIGGERLEVEL (*value*) (INTRAPARTITION QUEUES ONLY)
Displays the number of requests for output to a queue that must accrue before automatic transaction initiation (ATI) occurs. The number can be between 0 and 32767.

You can reset this value by overtyping it with a different value.

TYPE
Displays the type of this transient data queue.
INDIRECT
The queue type is indirect. The name of the final target queue is shown in the next field.
EXTRA
The queue type is extrapartition.
INTRA
The queue type is intrapartition.
REMOTE
The queue type is remote.

USERID
Displays the 8-byte identifier associated with the intrapartition queue intended for ATI.

CEMT INQUIRE TEMPSTORAGE

Retrieve information about storage that is used by main temporary storage queues in the CICS region.

Description

The **CEMT INQUIRE TEMPSTORAGE** command returns information about the CICS region storage that main temporary storage queues use, and the maximum amount that is available for their use.

Input

Press the Clear key to clear the screen. To start this transaction, type `CEMT INQUIRE TEMPSTORAGE` (or suitable abbreviations for the keywords). The status is displayed.

To change attributes, overtype your changes on the INQUIRE screen after tabbing to the appropriate field (see "Overtyping a display" on page 117).

If you change the **TSMAINLIMIT** value, check your current setting for the z/OS parameter **MEMLIMIT**. **MEMLIMIT** limits the amount of 64-bit storage that the CICS address space can use. Your setting for **TSMAINLIMIT** must not be greater than 25% of the **MEMLIMIT** value.

Use the CEMT command **INQUIRE DSAS** or **INQUIRE SYSTEM** to find the value of **MEMLIMIT** that currently applies to the CICS system.

Displayed fields

TSMAININUSE(*value*)
Displays how much storage is currently used by main temporary storage queues. The value is displayed in bytes, megabytes followed by the letter M, or gigabytes followed by the letter G.

TSMAINLIMIT(*value*)
Displays the current setting for the maximum amount of storage that CICS makes available for main temporary storage queues to use. The value is displayed either in gigabytes, or to the nearest number of megabytes.

If you decrease the TSMAINLIMIT setting, CICS attempts to maintain at least 25% free space in allowed storage above current utilization, so that temporary storage write requests do not reach TSMAINLIMIT too rapidly. The value is set as follows:

- If there is currently less than 25% free space, TSMAINLIMIT remains unchanged.
- If at least 25% of the new limit will be free space, the setting is decreased to the value that you choose.
- If less than 25% of the new limit would be free space, the setting is decreased to the current utilization plus 33% of that utilization.

If you increase the TSMAINLIMIT setting, the value is set as follows:

- If the new value is not greater than 25% of the value of the z/OS parameter **MEMLIMIT**, the value that you choose is set.
- If the new value is greater than 25% of the **MEMLIMIT** value, TSMAINLIMIT remains unchanged.

CEMT INQUIRE TERMINAL

Retrieve information about terminals.

In the CICS Explorer, the Terminals operations view provides a functional equivalent to this command.

Description

INQUIRE TERMINAL returns information about a named terminal that is present in the terminal control table (TCT).

In an SNA environment, a terminal is the CICS representation of an SNA logical unit with which CICS can be in communication. An SNA environment is one in which CICS is using z/OS Communications Server as the access method for communication with SNA logical units.

This logical unit can be:

- A physical terminal (such as a 3277)

- A function of a cluster controller, such as a 3790 Type 2 batch function
- An IP interconnectivity (IPIC) session, intersystem communication (ISC) session, or interregion communication (IRC) session that has been generated by the CEDA transaction
- A remote terminal
- A remote session; that is, a session of a remote connection

If you are familiar with network names, you might prefer to use the command that uses these names. See "CEMT INQUIRE NETNAME" on page 271.

The terminal entry displayed can also be an MVS console.

The message `TEMP OUT SERVICE` can be displayed if recovery is in progress for the terminal. The `TEMP OUT SERVICE` indicator is reset by refreshing the screen, or by overtyping the **INS** field with `INS` or `OUT`. If `TEMP OUT SERVICE` continues to be displayed, the cause of the indicator must be investigated.

Input

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT INQUIRE TERMINAL`. You get a display that lists the status.
- Type `CEMT INQUIRE TERMINAL` followed by the other attributes that are necessary to limit the range of information that you require. So, for example, if you enter `CEMT INQUIRE TERMINAL i at`, the resulting display shows you the details of only those terminals that are in service and available for use. Iif you enter `CEMT INQUIRE TERMINAL rem(cicr)`, the display shows you only remote terminals (or remote ISC or IRC sessions) owned by the system that the local CICS knows as CICR.

You can use these methods to change various attributes:

- Overtype your changes on the INQUIRE screen after using tab to go to the appropriate field. See "Overtyping a display" on page 117.
- Use the CEMT SET TERMINAL command.

(*value***)**
Is a terminal identifier (1 - 4 characters) as specified in an installed terminal definition.

If the terminal name is, for example, S201, code this option in the following way:

```
CEMT INQUIRE TERMINAL(S201)
```

ALL
Is the default. For example, if you inquire about terminals, you receive information about all terminals, unless you specify a terminal identifier, a terminal class identifier, or a system identifier.

CLASS (*value***)**
Is the 1- or 2-character suffix of a terminal list table (TLT).

If you do not specify a termid or class identifier, by default you receive information about all the terminals.

Sample screen

```
 IN TERM
 STATUS:  RESULTS - OVERTYPE TO MODIFY
  Ter(CBRF)           Pri( 000 ) Pag Ins Ati Tti Rte
     Net(CBRF    )                     Rem(CBR ) Rna(CBRF)
  Ter(CERR)           Pri( 000 ) Pag Ins     Tti Loc
     Con(*ERRCON*    )
  Ter(CTSO)           Pri( 000 ) Pag Ins     Tti Loc
     Con(PKING       )
  Ter(KING)           Pri( 000 ) Pag Ins     Tti Loc
     Con(PRUEKING    )
  Ter(RNAL)           Pri( 000 ) Pag Ins     Tti Loc
     Con(INTERNAL    )
  Ter(S225) Tra(CEMT) Pri( 000 ) Pag Ins Ati Tti Loc
     Net(IGCS225 ) Acq     Tas(0000068)           Nqn(GBIBMIYA.IGCS225 )
  Ter(S280) Tra(CEMT) Pri( 000 ) Pag Ins Ati Tti Loc
     Net(IGCS280 ) Acq     Tas(0000063)           Nqn(GBIBMIYA.IGCS280 )
  Ter(TC12) Tra(CEMT) Pri( 000 ) Pag Ins Ati Tti Loc
     Net(IYCWTC12) Acq     Tas(0000072)           Nqn(GBIBMIYA.IYCWTC12)
  Ter(-AAZ)           Pri( 000 ) Aut Ins Ati Tti Ses
     Net(TMPLATE1) Rel                   Rem(CBPS)
  Ter(-AA0)           Pri( 000 ) Aut Ins Ati Tti Ses
     Net(TMPLATE1) Rel Cre               Rem(CBPS)
+ Ter(-AA1)           Pri( 000 ) Aut Out Ati Tti Ses
     Net(TMPLATE1) Rel Cre               Rem(CBPS)
```

Figure 106. CEMT INQUIRE TERMINAL screen

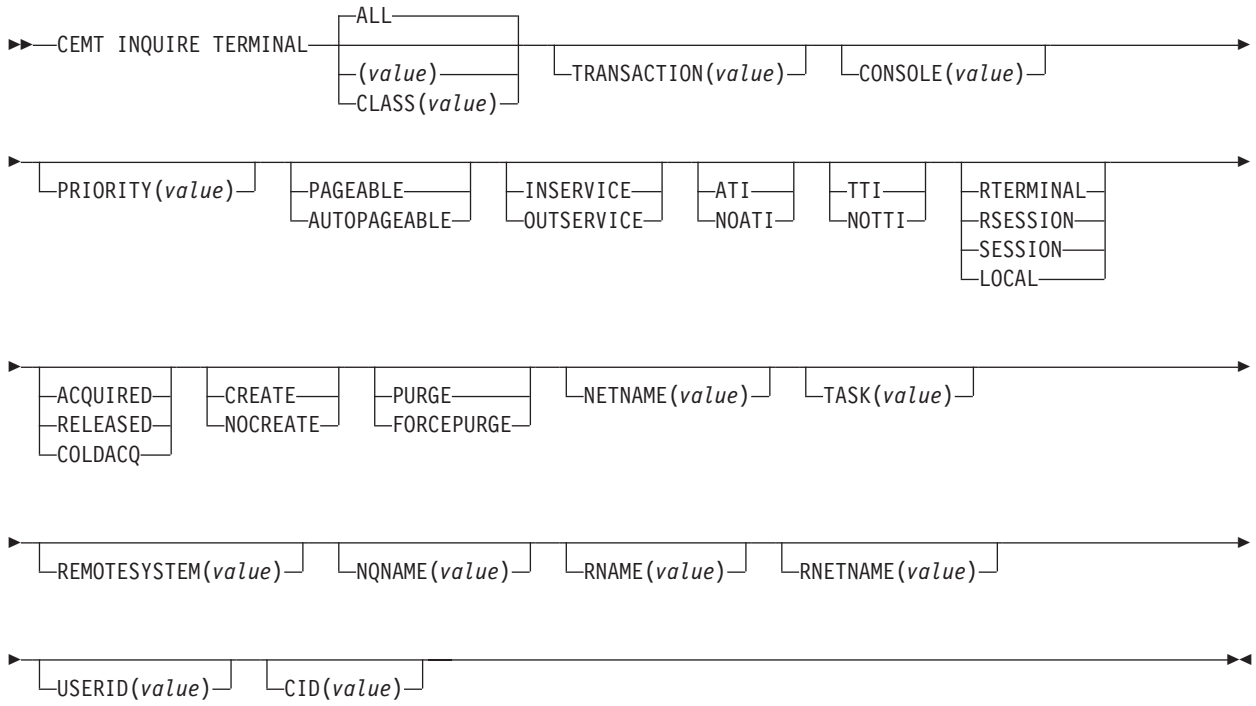
Note: There are blank fields on the screen where a value does not apply or is 'negative'; that is, the value begins with 'No'. To modify these fields, locate them by using tab (they are shown in the same sequence as in the expanded format), and overtype with input valid for that field. You might find it more convenient to use the expanded format when setting one of these values. Negating is easier because you are overtyping a field with its negative. Doing this prevents the result from being displayed.

Place the cursor against a specific entry in the list and press Enter; CICS shows an expanded format as shown in Figure 107.

```
 IN TERM
 RESULT - OVERTYPE TO MODIFY
   Terminal(bell)
   Transaction()
   Priority( 000 )
   Pagestatus( Pageable )
   Servstatus( Inservice )
   Atistatus( Noati )
   Ttistatus( Tti )
   Nature(Local)
   Purgetype(              )
   Netname()
   Console(BELL)
   Termstatus(              )
   Createsess(              )
   Task(00000)
   Remotesystem()
   Nqn(GBIBMIYA.IYCQTC03)
   Rname()
   Rnetname()
+  Cid()
```

Figure 107. The expanded display of an individual entry

CEMT INQUIRE TERMINAL



Displayed fields

AIDCOUNT (*data area*)
Returns a fullword binary field giving the number of Automatic Initiate Descriptors (AIDs) queued for the specified terminal. If there are no AIDs, then an AIDCOUNT value of 0 is returned. The count represents the number of AIDs on the CICS system that issued the command.

ATISTATUS
Displays whether the terminal is available for use by transactions that are automatically initiated from CICS or, if the terminal is an ISC session, by transactions that are using this session as an alternative way to communicate with another system. The values are:
ATI The terminal is available for use.
NOATI
The terminal is not available for use.

Note:

1. You can reset this value by overtyping it with a different value.
2. You cannot define a terminal with both NOATI and NOTTI.

CID (*value*)
Displays an 8-character correlation-identifier that is set only for sessions, and only after a session has been acquired. The value relates to the previously acquired session if there was one, if the session is not acquired. CID is set in the following ways:

- For LU6.2 sessions, it is an 8-character token that is common to the two end sessions that are connected together.

- For MRO sessions, it is set to the termid of the session on the system to which this session is connected.
- For LU6.1 sessions to IMS, it is set to the value of NETNAMEQ as defined or negotiated.
- For other LU6.1 sessions, it is set to the value supplied by the other end of the 6.1 session.

Using CID, you can relate the two parts of an LU6.2, MRO, or LU61 conversation:

- For LU6.2, you can inquire on the same CID.
- For MRO, you can locate the session with the terminal name.
- For LU6.1 to IMS, you can inquire on the netname.
- For other LU6.1, you can use the name to inquire on the connected system.

CONSOLE (*value***)**
Displays, for an MVS console only, a 12-byte string that contains the identifier of the console, in two parts. If the device is not a console, CICS returns 12 blanks.

If the console is autoinstalled, or is defined explicitly with a console name, the name is returned in the first 8 bytes, and the last 4 bytes are blank.

If the console is defined by a numeric identifier, the string is divided into two parts, separated by a period (.) in the ninth byte position. The 12-byte string contains the following information:

- The first 8 bytes contain the MVS console name, if it is known, or the string `*UNKNOWN` if it is not known.
- A period separating the name from the numeric ID.
- The last 3 bytes contain the numeric console ID.

CREATESESS (*z/OS Communications Server only***)**
Displays whether the terminal can be acquired automatically by ATI transactions. You cannot specify this field for IRC sessions. The values are:

CREATE
If the terminal is not in session, CICS acquires it if it is needed to satisfy an ATI request.

NOCREATE
If the terminal is not in session, CICS does not acquire it to satisfy an ATI request. A session must be started by, for example, a logon request or a **CEMT SET TERMINAL ACQUIRED** command before the ATI request can be satisfied.

If NOCREATE is set for an LU6.1 ISC session and no allocatable sessions remain, the connection is placed OUTSERVICE.

You can reset this value by overtyping it with a different value.

NATURE
Displays the nature of the terminal; that is, whether it is a physical device or a session; and whether it is local or remote. The values are:

RTERMINAL
The terminal is a remote device.

RSESSION
The terminal is a session of a remote connection.

SESSION
The terminal is an ISC or IRC session.

LOCAL
The terminal is directly attached to this system and is a physical device or a function of a cluster controller.

NETNAME (*value*)
Shows an 8-character network name.

For a physical terminal, the Netname is the name by which this terminal is known to z/OS Communications Server.

For ISC sessions, it is the name by which the session (or group of sessions) is known to z/OS Communications Server.

For IRC sessions, it is the name used by the connected region to log on to the interregion communication program (DFHIRP).

For a remote terminal, it is the name by which the terminal is known to the z/OS Communications Server in the remote region. (For a remote terminal routed from a pre-CICS TS for z/OS, Version 4.2 region, Netname is blank.)

If the netname is a z/OS Communications Server LU alias, it is different from the netname component of the NQNAME, which always contains the real netname.

NQNAME (*value*)
Displays the 17-character network-qualified name.

Remote terminals do not have an NQNAME value.

If the status (Termstatus) is RELEASED, NQNAME displays the netname.

PAGESTATUS
Displays whether pages after the first in a series are written to the terminal on request from the operator or automatically. The values are:
PAGEABLE
Pages are written on request.
AUTOPAGEABLE
Pages are written automatically.

You can reset this value by overtyping it with a different value.

PRIORITY (*value*)
Displays a 3-character string identifying the priority of a terminal relative to other terminals. The priority of a task is the sum of the transaction priority, the terminal priority, and the operator priority. Priority has no meaning for terminals that are ISC sessions being used as alternative facilities.

The value is in the range 0 - 255, where 255 is the highest priority.

You can reset this value by overtyping it with a different value.

PURGETYPE (*input only field*)
Specifies whether transactions running with the named terminal are to be purged. The values are:
PURGE
Transactions are stopped only if system and data integrity can be maintained. A transaction is purged if its definition specifies SPURGE=NO.
FORCEPURGE
Transactions are to be purged immediately. This can lead to unpredictable results and are for use in exceptional circumstances.

REMOTESYSTEM (*value*)
Displays the first four characters of a connection, if the subject of the inquiry is

a remote terminal. The named connection can be either a connection entry that links toward the terminal-owning region (TOR), or an indirect connection that provides the netname of the TOR.

Otherwise, this field is blank.

RNAME (*value***)**
Displays the 4-character name by which this terminal is known in the system in which it is defined (the TOR). RNAME applies only to terminals defined as remote; for others, blanks are displayed.

RNETNAME (*value***)**
Displays the 8-character netname of the owning TOR, if the subject of the inquiry is a remote terminal. If this field is blank and the terminal is remote, the indirect connection, displayed in REMOTESYSTEM, contains the netname of the TOR.

SERVSTATUS
Displays whether the terminal is available for use. These are the values:
INSERVICE
The terminal is available for use. For z/OS Communications Server, Ins(ervice) means that the terminal can be ACQUIRED. For IRC sessions, Ins(service) means that the connection to the MRO partner is INSERVICE.
OUTSERVICE
The terminal is not available for use. Setting a terminal Out(service) means that the terminal can no longer be used by transactions. If PURGE or FORCEPURGE is also specified, any transaction that uses the terminal is stopped abnormally. If PURGE or FORCEPURGE is not specified, the transaction ends normally, but no more transactions can use the terminal.

For z/OS Communications Server, setting a terminal Out(service) also causes it to be released and the operator to be signed off, either immediately or when the current transaction has ended. For IRC sessions, Out(service) means that the connection to the MRO partner is OUTSERVICE.

In an LU6.1 ISC session, the connection is set Out(service) if there are no allocatable sessions are left.

For physical terminals and LU6.1 sessions, you can reset this value by overtyping it with a different value.

TASK (*value***)**
Displays the task number of the transaction that is running on this terminal.

TERMINAL (*value***)**
Indicates that this panel relates to a TERMINAL inquiry and displays a 4-character terminal identifier as defined in an installed terminal definition. This option includes all terminals and sessions, but not logical device codes (LDCs), model TCTTEs, mode groups, or system entries. See also **Net**name.

TERMSTATUS (*z/OS Communications Server only***)**
Displays whether CICS is in session with the logical unit represented by this terminal. Theses are the values:
ACQUIRED
CICS is in session with the logical unit represented by the terminal.
RELEASED
CICS is not in session with the logical unit represented by the terminal. If you set this option to RELEASED, a session is stopped immediately

if you also specify the PURGE option; otherwise, the session is ended when the current active transaction is completed.

For physical terminals and LU6.1 sessions, you can reset this value by overtyping it with a different value or with:

COLDACQ
CICS is in session with the logical unit represented by the terminal, where no resynchronization is required.

TRANSACTION (*value***)**
Displays a 4-character string identifying the name of the transaction currently being processed with this terminal as its principal facility or as a secondary facility.

TTISTATUS
Displays whether the terminal can be used by the transactions that are initiated from this terminal. The values are:

TTI This terminal can be used by transactions.

NOTTI
This terminal cannot be used by transactions.

Note:

1. You can reset this value by overtyping it with a different value.
2. A terminal cannot be defined with both NOATI and NOTTI.

USERID (*data-area***)**
Returns the 8-character identifier of the user signed on at this terminal or session.

If there is no signed-on user, the default user ID, as specified in the **DFLTUSER** system initialization parameter, is returned.

CEMT INQUIRE TRANSACTION

Retrieve information about transactions.

In the CICS Explorer, the Transactions operations view provides a functional equivalent to this command.

Description

The **INQUIRE TRANSACTION** command returns information about transaction definitions.

The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT INQUIRE TRANSACTION` (or suitable abbreviations for the keywords). The resulting display lists the current status.
- Type `CEMT INQUIRE TRANSACTION` (or suitable abbreviations for the keywords), followed by the attributes that are necessary to limit the range of information that you require. For example, if you enter `cemt i trans en pu`, the resulting display shows the details of only those transactions that are enabled and system-purgeable.

You can change various attributes in the following ways:

- Overtype your changes on the INQUIRE screen after tabbing to the appropriate field. See "Overtyping a display" on page 117.
- Use the CEMT SET TRANSACTION command.

(*value***)**
A 1- to 4-character transaction identifier. Only transactions that have been defined in the CICS system definition (CSD) file and installed on the running CICS system are accessible through CEMT.

ALL
The default.

CLASS(*value***)**
Is the 2-character suffix of a transaction list table (XLT).

Sample screen

```
 IN TRAN
 STATUS:  RESULTS - OVERTYPE TO MODIFY
  Tra(AADD) Pri( 001 ) Pro(DFH$AALL) Tcl( DFHTCL00 ) Ena Pur
     Prf(DFHCICST) Uda Bel Iso
  Tra(ABRW) Pri( 001 ) Pro(DFH$ABRW) Tcl( DFHTCL00 ) Ena Pur
     Prf(DFHCICST) Uda Bel Iso
  Tra(ADYN) Pri( 001 ) Pro(DFH99   ) Tcl( DFHTCL00 ) Ena Pur
     Prf(DFHCICST) Uda Bel Iso
  Tra(AINQ) Pri( 001 ) Pro(DFH$AALL) Tcl( DFHTCL00 ) Ena Pur
     Prf(DFHCICST) Uda Bel Iso
  Tra(AMNU) Pri( 001 ) Pro(DFH$AMNU) Tcl( DFHTCL00 ) Ena Pur
     Prf(DFHCICST) Uda Bel Iso
  Tra(AORD) Pri( 001 ) Pro(DFH$AREN) Tcl( DFHTCL00 ) Ena Pur
     Prf(DFHCICST) Uda Bel Iso
  Tra(AORQ) Pri( 001 ) Pro(DFH$ACOM) Tcl( DFHTCL00 ) Ena Pur
     Prf(DFHCICST) Uda Bel Iso
  Tra(AREP) Pri( 001 ) Pro(DFH$AREP) Tcl( DFHTCL00 ) Ena Pur
     Prf(DFHCICST) Uda Bel Iso
+ Tra(AUPD) Pri( 001 ) Pro(DFH$AALL) Tcl( DFHTCL00 ) Ena Pur
     Prf(DFHCICST) Uda Bel Iso
```

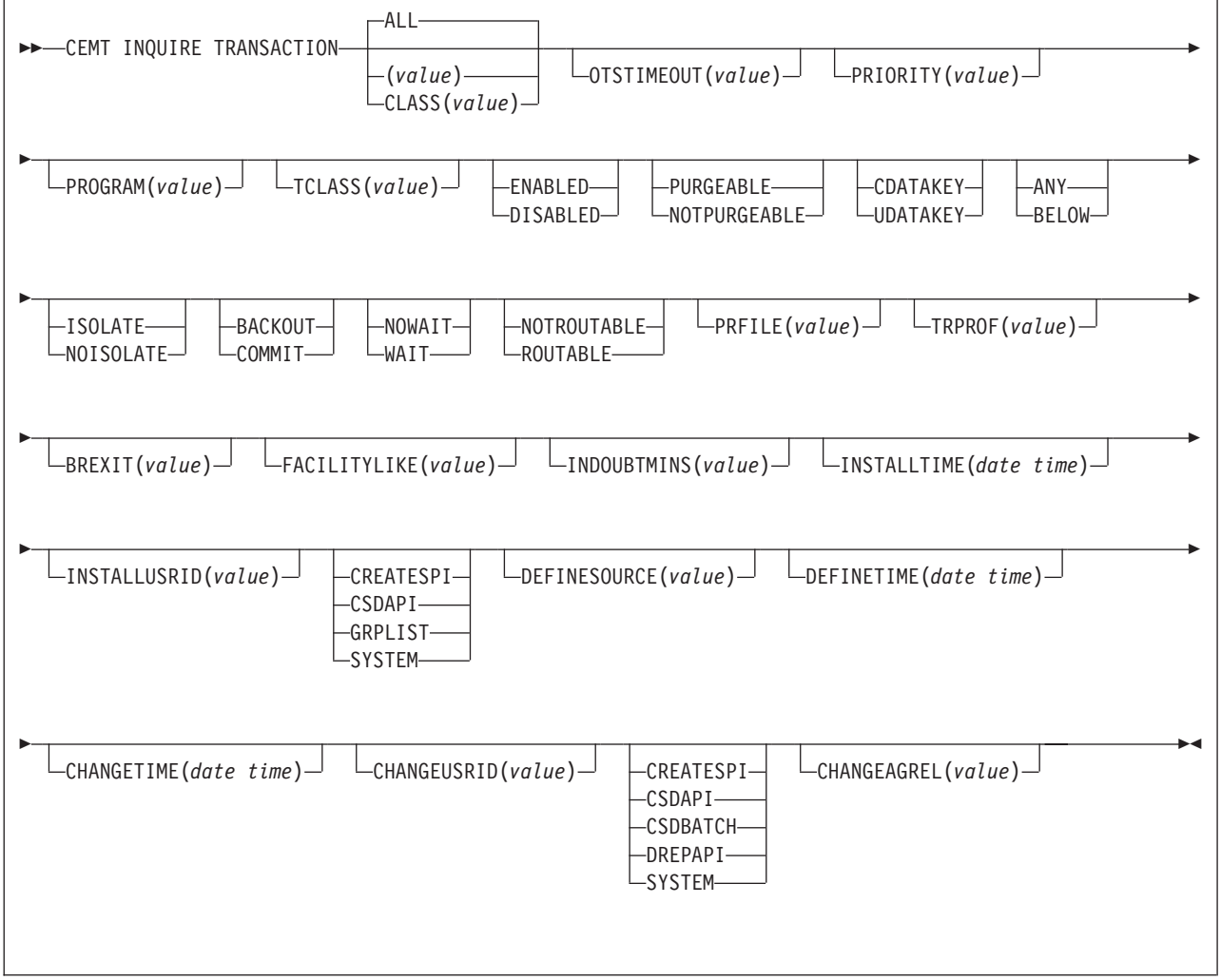
Figure 108. CEMT INQUIRE TRANSACTION screen

If you place the cursor against a specific entry in the list and press ENTER, CICS displays an expanded format, as shown in Figure 109 on page 351.

```
IN TRAN
RESULT - OVERTYPE TO MODIFY
  Transaction(AADD)
  Priority( 001 )
  Program(DFH$AALL)
  Tclass( DFHTCL00 )
  Status( Enabled )
  Routing(Dynamic)
  Routestatus(Notroutable)
  Purgeability( Purgeable )
  Prfile(DFHCICST)
  Taskdatakey(Udatakey)
  Taskdataloc(Below)
  Isolatest(Isolate)
  Trprof()
  Indoubt
  INdoubtwait
  Brexit
  Facilitylike
  Otstimeout(12000)
```

Figure 109. The expanded display of an individual entry

CEMT INQUIRE TRANSACTION



Displayed fields

BREXIT(*value***)**
Returns the 8-character name of the bridge exit defined by the BREXIT parameter of the named transaction resource definition.

If BREXIT is not defined, blanks are returned.

CHANGEAGENT(*value***)**
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

SYSTEM
The resource definition was last changed by the CICS or CICSPlex system.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

DEFINESOURCE(*value***)**
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

FACILITYLIKE(*value***)**
Returns the 4-character name of the terminal defined by the FACILITYLIKE parameter of the PROFILE associated with the named transaction resource definition.

INDOUBT
Displays the action, based on the ACTION attribute of the TRANSACTION

resource definition, to be taken if the CICS region fails or loses connectivity with its coordinator while a unit of work is in the indoubt period.

The action depends on the values returned in Indoubtwait and Indoubtmins; if Indoubtwait returns WAIT, the action is not normally taken until the time returned in Indoubtmins expires. For exceptions to this rule, see Indoubtwait.

The values are as follows:

BACKOUT
All changes made to recoverable resources are to be backed out.

COMMIT
All changes made to recoverable resources are to be committed, and the unit of work marked as completed.

INDOUBTMINS(*value***)**
Displays the length of time, in minutes, after a failure during the indoubt period, before the transaction is to take the action returned in the Indoubt field. The returned value is valid only if the unit of work is indoubt and Indoubtwait returns WAIT.

See also Indoubt and Indoubtwait.

INDOUBTWAIT
Displays, based on the WAIT attribute of the TRANSACTION definition, how CICS is to respond if a failure occurs while a unit of work (UOW) is in an indoubt state. The values are as follows:

NOWAIT
The UOW is not to wait, pending recovery from the failure. CICS is to take immediately whatever action is specified on the ACTION attribute of the TRANSACTION definition.

WAIT
The UOW is to wait, pending recovery from the failure, to determine whether recoverable resources are to be backed out or committed.

Even if Indoubtwait returns WAIT, aspects of the UOW might force CICS to take an immediate decision; that is, to take immediately the action specified on the ACTION attribute of the transaction definition. Such processing can happen if, for example, the UOW contains one of these sessions:

- Subordinate LU6.1 sessions
- Subordinate MRO sessions to pre-CICS Transaction Server for z/OS systems.

For further information about the meaning of the ACTION and WAIT attributes of the TRANSACTION definition, see the *CICS Resource Definition Guide*.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

SYSTEM
The resource was installed by the CICS or CICSPlex SM system.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

ISOLATEST
Displays whether the user-key task-lifetime storage is isolated from the user-key programs of other transactions.

ISOLATE
The user-key task-lifetime storage of the transaction is accessible only by the user-key programs of its own task. The user-key task-lifetime storage of the transaction is isolated from all the user-key programs of all other tasks.

NOISOLATE
The user-key task-lifetime storage of the transaction is accessible by its own programs, and also by user-key programs of other transactions defined with the ISOLATE(NO) option.

OTSTIMEOUT(*value***)**
Displays a fullword data-area containing the default period in seconds that an OTS transaction, created in an EJB environment running under this CICS transaction, is allowed to execute before sync point.

PRFILE(*value***)**
Displays the name of the profile definition that defines additional options associated with this transaction.

PRIORITY(*value***)**
Displays a value indicating the priority of a transaction relative to other transactions. When a transaction is running as a CICS task, the priority of a task is the sum of the transaction priority, the terminal priority, and the operator priority.

You can reset this value by overtyping it with a different value. The value is in the range 0 - 255, where 255 is the highest priority.

PROGRAM(*value***)**
Displays an 8-character string identifying the name of the first program to run when this transaction is started.

PURGEABILITY
Displays whether the transaction is purgeable in system stall conditions. The values are as follows:

PURGEABLE
The transaction is system-purgeable. This value relates to the SPURGE parameter on the transaction resource definition and indicates that CICS can purge the transaction in a deadlock timeout. See the *CICS Resource Definition Guide* for information about the SPURGE and DTIMOUT parameters on a transaction resource definition.

NOTPURGEABLE
The transaction cannot be purged.

You can reset this value by overtyping it with a different value.

ROUTSTATUS
Displays whether, if the transaction is the subject of an eligible EXEC CICS START command, it will be routed using the enhanced routing method. The values are as follows:

NOTROUTABLE
If the transaction is the subject of a START command, it is routed using the "traditional" method.

ROUTABLE
If the transaction is the subject of an eligible START command, it is routed using the enhanced method.

For details of the enhanced and "traditional" methods of routing transactions invoked by EXEC CICS START commands, see the *CICS Intercommunication Guide*.

STATUS
Displays whether the transaction is available for use.

ENABLED
The transaction is available for use.

DISABLED
The transaction is not available for use.

A disabled transaction does not prevent a START command that names this transaction from being shipped to a remote region. When a task is attached for the requested transaction, CICS checks that the transaction is enabled in the remote region.

Note: You can reset this value by overtyping it with a different value.

TASKDATAKEY
Displays the storage key in which CICS obtains all storage for use by the transaction. This storage includes the task life-time storage, that is the transaction work area (TWA) and the EXEC interface block (EIB), and the storage that CICS obtains on behalf of programs that run under the transaction.

The values are as follows:

CDATAKEY
CICS obtains storage for the transaction from CICS-key storage. Application programs that run in CICS key have read-write access to this storage, but user-key programs have read-only access.

UDATAKEY
CICS obtains storage for the transaction from user-key storage. Application programs that run in any key have read-write access to this storage.

See the description of the TASKDATAKEY parameter on the transaction resource definition in the *CICS Resource Definition Guide*.

TASKDATALOC
Displays whether certain CICS control blocks, including EIB and TWA, for a transaction are acquired above or below 16 MB.

ANY The transaction accepts task-related data anywhere.

BELOW
The transaction requires any task-related data (TWA and EIB plus any internal control blocks) to be located below 16 MB.

TCLASS(*value*)
Displays an 8-character string identifying the name of the transaction class to which the transaction belongs. If the transaction does not belong to a class, DFHTCL00 is returned.

You can reset this value by overtyping it with a different value.

To remove a transaction from its TCLASS, set this field to DFHTCL00. An added or changed TCLASS *must* be one that has already been defined.

TRANSACTION(*value*)
Indicates that this panel relates to a TRANSACTION inquiry and displays a 4-character transaction identifier. Only transactions that have been defined in the CICS system definition (CSD) file and installed on the running CICS system are accessible through CEMT.

TRPROF(*value*)
Displays the name of the transaction routing profile that defines additional options associated with this transaction if it is defined as a remote transaction.

CEMT INQUIRE TRDUMPCODE

Retrieve information about transaction dump codes.

Description

The **INQUIRE TRDUMPCODE** command shows the current settings of the entries in the transaction dump table. See the *CICS Problem Determination Guide* for information about using dumps.

Note: Do not use both the SHUTDOWN and MAXIMUM options in the same command.

This command returns entries that were added to the dump table by a user application program or CEMT command, and also entries that CICS added to the dump table through dump requests. Be aware that the entries that CICS added to the dump table are temporary, and they are not restored during a warm start, or during a cold or initial start. Entries added by a user application program or CEMT command are restored during a warm start, but not during a cold or initial start.

Input

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT INQUIRE TRDUMPCODE` (or suitable abbreviations for the keywords). The resulting display lists the current status.
- Type `CEMT INQUIRE TRDUMPCODE` (or suitable abbreviations for the keywords), followed by the attributes that are necessary to limit the range of information that you require. For example, if you enter `cemt i trd sy sh`, the resulting display shows the details of only those transaction dump codes for which system dumps are taken and for which the CICS system shuts down.

you can change various attributes in the following ways:

- Overtype your changes on the INQUIRE screen after tabbing to the appropriate field (see "Overtyping a display" on page 117).
- Use the **CEMT SET TRDUMPCODE** command.

(*value***)**
A 4-byte transaction dump code. See *CICS Messages and Codes Vol 1* for information on transaction dump codes.

ALL
The default.

Sample screen

```
IN TRDU
STATUS:  RESULTS - OVERTYPE TO MODIFY
 Trd(ATNI) Tra         Loc Max( 999 ) Cur(0002)
 Trd(ERRS) Tra         Loc Max( 999 ) Cur(0001)
```

Figure 110. CEMT INQUIRE TRDUMPCODE screen

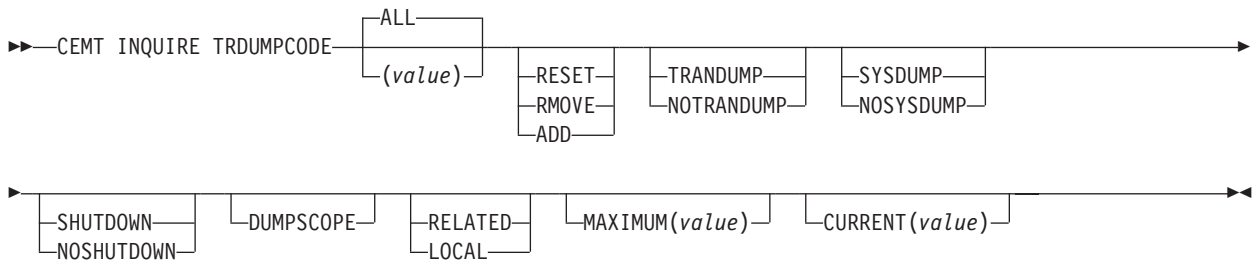
Note: When a value does not apply, or is negative (the value begins with No), the fields on the screen are blank. To modify these fields, locate them by tabbing (they appear in the same sequence as in the expanded format), and overtype with input valid for that field. You might find it more convenient to use the expanded format when setting one of these values.

If you place the cursor against a specific entry in the list and press ENTER, CICS displays an expanded format as shown in Figure 111.

```
IN TRDU
RESULT - OVERTYPE TO MODIFY
  Trdumpcode(ATNI)
  Trandumping( Trandump )
  Sysdumping( Nosysdump )
  Shutoption( Noshutdown )
  Dumpscope( Local )
  Maximum( 999 )
  Current(0002)
  Action(             )
```

Figure 111. The expanded display of an individual entry

CEMT INQUIRE TRDUMPCODE



Displayed fields

ACTION (*input only field***)**
Displays whether the specified transaction dump code is reset, removed from the dump code table, or added to the dump code table. The values are as follows:

RESET
Reset the current count of dump calls for this dump code to zero.

REMOVE
Remove the specified transaction dump code from the dump code table.

ADD Add the specified transaction dump code to the dump code table.

CURRENT(*value***)**
Displays the number of dump calls that have been made for this transaction dump code since the number was last reset to zero. The value of CURRENT is limited to four characters; if there were more than 9999 dump calls for this transaction dump code since it was last reset, CURRENT contains only the four least-significant figures of the value.

The CURRENT value can be reset explicitly by using a SET SYDUMPCODE RESET command, or automatically when CICS shuts down.
You can reset this value by overtyping it with a different value.

DUMPSCOPE
Displays whether a system dump request is sent to MVS images in the sysplex that are running XCF/MRO connected CICS regions related to the CICS region on which the dump is initiated.

Note: A related CICS region is one in which the unit of work identifiers, in the form of APPC tokens, of one or more tasks match those in the CICS region that issued the dump request.

RELATED
A system dump request is sent to other MVS images in the sysplex for this system dump code.

LOCAL
A system dump request is not sent to other MVS images in the sysplex for this system dump code.

You can reset this value by overtyping it with a different value.

MAXIMUM(*value***)**
Displays the largest number of dump calls for this dump code that result in dumps being produced. The value is in the range 0 - 999 and can be specified by using TRDUMAX in the SIT parameter. A value of 999 means the default, no limit.

SHUTOPTION
Displays whether the CICS system shuts down after an error that corresponds to this dump code occurs. The values are as follows:

SHUTDOWN
The CICS system shuts down.

NOSHUTDOWN
The CICS system does not shut down.

You can reset this value by overtyping it with a different value.

SYSDUMPING
Displays whether a system dump is produced for this transaction dump code. The values are as follows:

SYSDUMP
Produce a system dump.

NOSYSDUMP
Do not produce a system dump.

You can reset this value by overtyping it with a different value.

TRANDUMPING
Displays whether a transaction dump is produced for this transaction dump code. The values are as follows:

TRANDUMP
Produce a transaction dump.

NOTRANDUMP
Do not produce a transaction dump.

You can reset this value by overtyping it with a different value.

TRDUMPCODE(*value***)**
Indicates that this panel relates to a TRDUMPCODE inquiry and displays a 4-character transaction dump code. See *CICS Messages and Codes Vol 1* for information on transaction dump codes.

CEMT INQUIRE TSMODEL

Retrieve information about temporary storage (TS) models.

In the CICS Explorer, the TS models operations view provides a functional equivalent to this command.

Description

The **CEMT INQUIRE TSMODEL** command returns information about temporary storage models.

The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT INQUIRE TSMODEL` (or suitable abbreviations for the keywords). The resulting display lists the status.
- Type `CEMT INQUIRE TSMODEL` (or suitable abbreviations for the keywords), followed by the attributes that are necessary to limit the range of information that you require. For example, if you enter `cemt i tsm main`, the resulting display shows the details of only temporary storage models for main temporary storage queues.

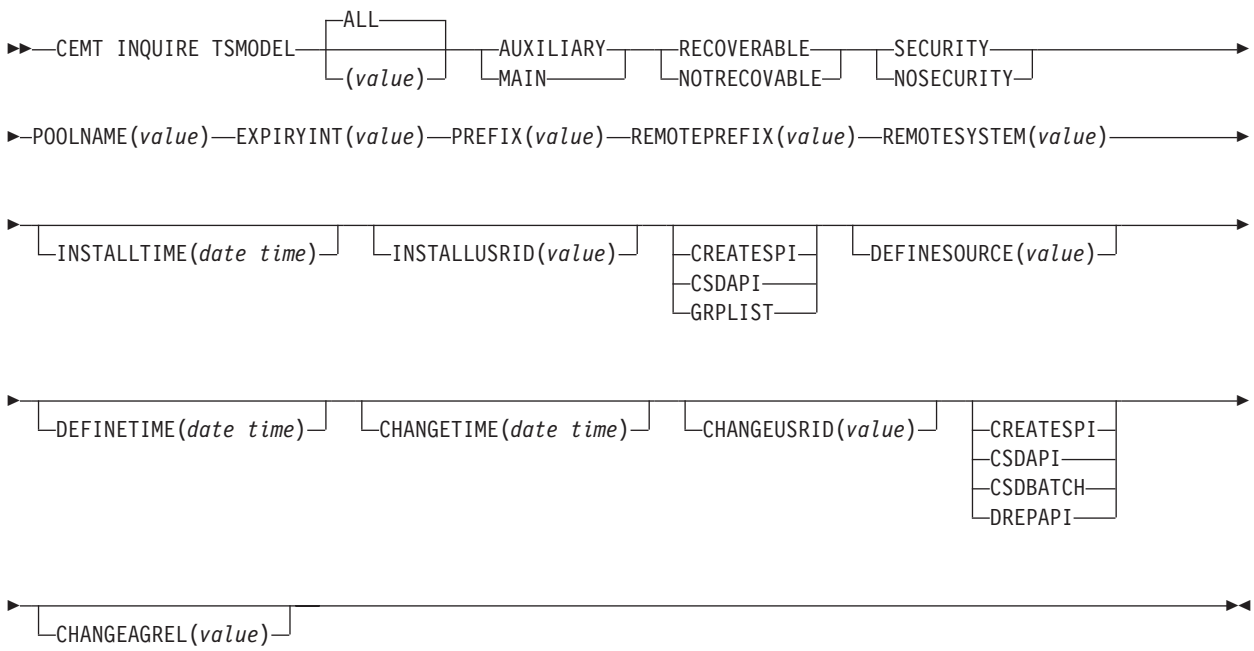
If you place the cursor against a specific entry in the list and press Enter, CICS displays an expanded format.

To change various attributes, overtype your changes on the INQUIRE screen after tabbing to the appropriate field (see Overtyping a display).

ALL
The default.

(*value***)**
The name of the temporary storage model for which information is requested.

CEMT INQUIRE TSMODEL



Displayed fields

CHANGEAGENT(*value***)**
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

DEFINESOURCE(*value***)**
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

EXPIRYINT(*value***)**
Displays the expiry interval, in hours, for temporary storage queues associated with this TS model. If a temporary storage queue is not referenced during its expiry interval, it becomes eligible to be deleted automatically by CICS. A value of zero means that no expiry interval applies to queues matching this model, so they are never eligible for automatic deletion. CICS does not apply an expiry interval to recoverable, remote, or shared temporary storage queues, or temporary storage queues created by CICS.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

LOCATION
Displays the location where temporary storage queues associated with this TS model reside. The values are:

AUXILIARY
TS queues associated with this TS model are held in the auxiliary temporary storage VSAM data set DFHTEMP, or in temporary storage pools in the coupling facility.

MAIN
TS queues associated with this TS model are held in main storage.

POOLNAME(*value***)**
Displays an 8-character shared pool name.

PREFIX(*value***)**
Displays the prefix for this model as a 16-character string.

RECOVSTATUS
Displays the recovery status of the temporary storage queue. The values are as follows:

RECOVERABLE
The temporary storage queue is recoverable.

NOTRECOVABLE
The temporary storage queue is not recoverable.

REMOTEPREFIX(*value***)**
Displays the remote prefix for this model as a 16-character string.

REMOTESYSTEM(*value***)**
Displays the remote system name for this model as a 4-character string.

SECURITYST
Displays the security status for this model. The values are as follows:

SECURITY
Security checking is performed for queue names matching this model.

NOSECURITY
Security checking is not performed for queue names matching this model.

TSMODEL(*value***)**
Indicates that this panel relates to a TSMODEL inquiry and displays the 8-character name of a temporary storage model.

Note: Nondisplayable characters appear as periods. You can use PF2 on the expanded panel to see the value in hexadecimal.

CEMT INQUIRE TSPOOL

Retrieve information about a temporary storage (TS) pool.

Description

The INQUIRE TSPOOL command returns information about a temporary storage (TS) pool.

Input

Press the Clear key to clear the screen. There are two ways to start this transaction:

- Type CEMT INQUIRE TSPOOL (or suitable abbreviations for the keywords). The resulting display lists the current status.
- Type CEMT INQUIRE TSPOOL (or suitable abbreviations for the keywords), followed by as many attributes as are necessary to limit the range of information that you require. For example, if you enter cemt i tsp connected, the resulting display lists the details of only those shared temporary storage pools that are connected.

To change various attributes, overtype your changes on the INQUIRE screen after tabbing to the appropriate field (see "Overtyping a display" on page 117).

ALL
The default.

(*value***)**
The name of the temporary storage pool for which information is requested.

Sample screen

```
I TSP
STATUS:  RESULTS
 Tsp(DFHWEB  ) Unc
```

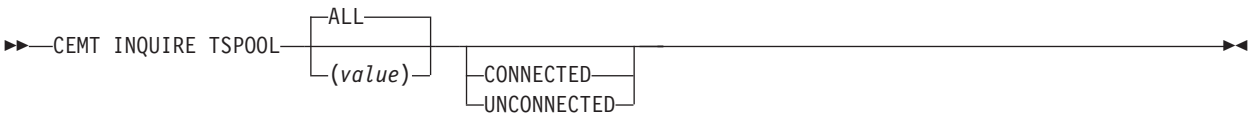
Figure 112. CEMT INQUIRE TSPOOL screen

If you place the cursor against a specific entry in the list and press ENTER, CICS displays an expanded format as shown in Figure 113.

```
I TSP
RESULT
  Tspool(DFHWEB)
  Connstatus(Unconnected)
```

Figure 113. The expanded display of an individual entry

CEMT INQUIRE TSPOOL



Displayed fields

CONNSTATUS
Displays the connection status of this TS pool. The values are:
CONNECTED
This TS pool is connected.
UNCONNECTED
This TS pool is not connected.

TSPOOL(*value***)**
Indicates that this panel relates to a TSPOOL inquiry and displays the 8-character name of a temporary storage pool.

Note: Nondisplayable characters appear as periods. You can use PF2 on the expanded panel to see the value in hexadecimal.

CEMT INQUIRE TSQUEUE / TSQNAME

Retrieve information about temporary storage queues by using the **CEMT INQUIRE TSQUEUE** or **CEMT INQUIRE TSQNAME** commands. In CEMT, you can use either command to operate on temporary storage queues (TS queues) with names up to 16 characters long.

In the CICS Explorer, the TS queues operations view provides a functional equivalent to this command.

Description

The **INQUIRE TSQUEUE** command returns information about temporary storage (TS) queues. The INQUIRE TSQUEUE command operates on all the temporary storage queues that exist in the CICS region. Temporary storage queues that CICS creates internally for its own use, for example, queues used by BMS, are included. You can identify the temporary storage queues that CICS creates for its own use by queue names that begin with the following character strings:

****** BMS paging
$$ BMS route
X'FA' to X'FF'
CICS
CEBR Default CEBR queue name
DF CICS
DFHM
Message cache for message-protected tasks
DFXX*xx*
CICS REQIDS (where *xx* is a hexadecimal value)

Input

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT INQUIRE TSQUEUE` (or suitable abbreviations for the keywords). The resulting display lists the status.
- Type `CEMT INQUIRE TSQUEUE` (or suitable abbreviations for the keywords), followed by the attributes that are necessary to limit the range of information that you require. For example, if you enter `cemt i tsq main`, the resulting display shows the details of only those temporary storage queues that are resident in main storage.

To change various attributes, over type your changes on the INQUIRE screen after tabbing to the appropriate field (see Overtyping a display).

ALL
The default.

(*value***)**
The name of the temporary storage queue for which information is requested.

SYSID(*value***)**
The system name that corresponds to the temporary storage pool name if the request is to be sent to a TS server.

This value is input-only, and must be specific; you cannot specify generic names for SYSID. The specified system name must be defined in a temporary

storage table (TST) in a TST TYPE=SHARED entry. If the SYSID is not found in a TST TYPE=SHARED entry, CICS returns a SYSID NOT FOUND error.

POOLNAME(*value***)**
Specifies the 8-character pool name of a temporary storage pool if the request is sent to a TS server. Use this option to view shared temporary storage queues that are defined by RDO resource definitions.

If the pool name is not found, CICS returns a POOLERR.

Browsing

If you place the cursor against a specific entry in the list of temporary storage queues and press Enter, CICS displays an expanded format.

If you place the cursor against a specific entry in the expanded list and type a `B` (uppercase or lowercase), CICS invokes a direct link to the program DFHEDFBR used by the CEBR transaction to enable you to browse the contents of this TS queue. See CEBR - temporary storage browse for details.

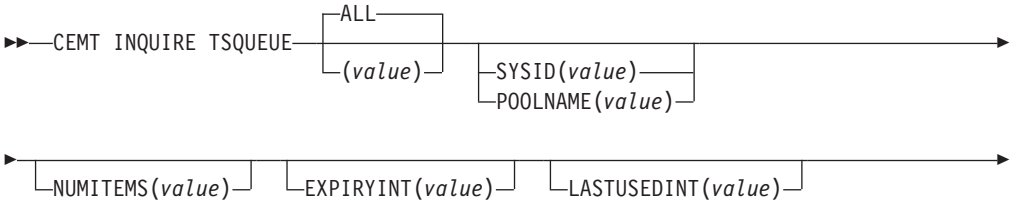
Security warning: Take care to avoid a security exposure. In this situation, access to the browse function is not controlled by transaction attach security checking using the **XTRAN** parameter in DFHSIT, as it is for the CEBR transaction. Instead, access is controlled by application program security checking using the **XPPT** parameter in DFHSIT.

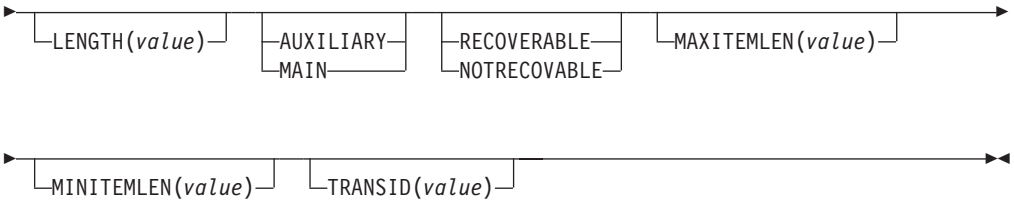
Not all TS queues visible on the CEMT INQUIRE TSQUEUE results panel can necessarily be browsed using CEBR. Queues resident in a shared TS pool, when the POOLNAME is specified on the INQUIRE TSQUEUE request, can sometimes not be browsed. Such queues can be browsed if there is a TSMODEL resource definition or temporary storage table (TST) entry mapping the queue name to the TS pool.

You can also browse queues in a shared pool by specifying a SYSID that maps to the shared pool using an entry in the temporary storage table (TST). However, the SYSID is not available to CEMT, so entering `B` next to the queue gives a NOT FOUND response from CEBR. If an appropriate SYSID is then provided by the user, the reference to the shared pool can be resolved.

If there is no suitable entry in the temporary storage table (TST) and no TSMODEL mapping the queue name to the shared pool, the queue cannot be browsed. CEBR indicates that the requested queue does not exist.

CEMT INQUIRE TSQUEUE/TSQNAME





Displayed fields

EXPIRYINT(*value*)
Displays the expiry interval, in hours, that is defined for the temporary storage queue in its TSMODEL resource definition. If the temporary storage queue is not referenced during the expiry interval, it becomes eligible to be deleted automatically by CICS.

A value of zero means that no expiry interval applies to the temporary storage queue, so it is never eligible for automatic deletion. In addition, the following types of temporary storage queues are never deleted automatically by CICS, even if a nonzero expiry interval is set in the matching TSMODEL resource definition:

- Queues in auxiliary temporary storage that are defined as recoverable.
- Queues in a remote CICS region.
- Queues that CICS creates for its own use.
- Temporary storage queues in shared temporary storage pools.

LASTUSEDINT(*value*)
Displays the elapsed time interval in binary seconds since the queue was last referenced.

LENGTH(*value*)
Displays the total length in bytes of all the items in the temporary storage queue.

The length of a queue item is the sum of the length of the user data plus 8 bytes for header information, rounded up.

- For main temporary storage, the length is rounded up to the boundary of the MVS storage subpool used to store it.
- For auxiliary temporary storage, the length is rounded to the next highest multiple of either 64 or 128. The control interval size of the temporary storage data set determines which is chosen. See Control interval size for auxiliary temporary storage.
- For shared queues, the lengths returned in MINITEMLEN, MAXITEMLEN, and FLENGTH, reflect the data length stored in the coupling facility. The data length includes any item control information, which consists of a 2-byte length prefix for each item.

For all types of queue, the maximum value returned is capped at 32767 (X'7FFF').

LOCATION
Displays where the temporary storage queue resides. The values are as follows:
AUXILIARY
The temporary storage queue is held in auxiliary temporary storage (the VSAM data set DFHTEMP), or in temporary storage pools in the coupling facility.

MAIN
The temporary storage queue is held in main storage.

MAXITEMLEN(*value*)
Displays the length in bytes of the largest item in the temporary storage queue.

The length of a queue item is the sum of the length of the user data plus 8 bytes for header information, rounded up.

- For main temporary storage, the length is rounded up to the boundary of the MVS storage subpool used to store it.
- For auxiliary temporary storage, the length is rounded to the next highest multiple of either 64 or 128. The control interval size of the temporary storage data set determines which is chosen. See Control interval size for auxiliary temporary storage.
- For shared queues, the lengths returned in MINITEMLEN, MAXITEMLEN, and FLENGTH, reflect the data length stored in the coupling facility. The data length includes any item control information, which consists of a 2-byte length prefix for each item.

For all types of queue, the maximum value returned is capped at 32767 (X'7FFF').

MINITEMLEN(*value*)
Displays the length in bytes of the smallest item in the temporary storage queue.

The length of a queue item is the sum of the length of the user data plus 8 bytes for header information, rounded up.

- For main temporary storage, the length is rounded up to the boundary of the MVS storage subpool used to store it.
- For auxiliary temporary storage, the length is rounded to the next highest multiple of either 64 or 128. The control interval size of the temporary storage data set determines which is chosen. See Control interval size for auxiliary temporary storage.
- For shared queues, the lengths returned in MINITEMLEN, MAXITEMLEN, and FLENGTH, reflect the data length stored in the coupling facility. The data length includes any item control information, which consists of a 2-byte length prefix for each item.

For all types of queue, the maximum value returned is capped at 32767 (X'7FFF').

NUMITEMS(*value*)
Displays the number of items in the temporary storage queue.

RECOVSTATUS
Displays the recovery status of the temporary storage queue. The values are as follows:
RECOVERABLE
The temporary storage queue is recoverable.
NOTRECOVABLE
The temporary storage queue is not recoverable.

TRANSID(*value*)
Displays the 4-character ID of the transaction that created the TS queue.

TSQUEUE(*value*)
Indicates that this panel relates to a TSQUEUE inquiry and displays the 16-character name of a temporary storage queue.

Note: Characters that cannot be displayed appear as periods. You can use PF2 on the expanded panel to see the value in hexadecimal.

CEMT INQUIRE UOW

Retrieve information about units of work (UOWs).

Description

The **INQUIRE UOW** command returns information about a named unit of work, or about all the UOWs currently in the system. It displays the state of the UOW (for example, INDOUBT) and whether it is active, waiting, or shunted.

If you suspect a problem with either a recoverable data set or a connection, you can use **INQUIRE UOW** to display UOWs that have been shunted because of a connection or data set failure. The command, in some cases, displays the name of the resource that caused the UOW to be shunted, plus the transaction, user, and terminal that started it.

Important: In an intercommunication environment, a unit of work can include actions by two or more connected systems. Such a unit of work is known as a *distributed* unit of work, because the resources to update are distributed across more than one system. A distributed unit of work is made up of two or more *local* units of work, each of which represents the work to be done on one of the participating systems.

The **INQUIRE UOW** command always returns information about *local* UOWs; that is, for a distributed UOW, it returns information only about the work required on the system on which the command is issued. You can assemble information about a distributed UOW by matching the network-wide UOW identifier returned in the NETUOWID field against the network-wide identifiers of local UOWs on other systems.

For further information about local and distributed UOWs, see the *CICS Intercommunication Guide*.

Input

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT INQUIRE UOW` (or suitable abbreviations for the keywords). The resulting display lists all the current UOWs.
- Type `CEMT INQUIRE UOW` (or suitable abbreviations for the keywords), followed by the attributes that are necessary to limit the range of information that you require. For example, if you enter `cemt i uow sh con`, the resulting display shows the details of only those UOWs that have been shunted because of the failure of a connection.

(*value***)**
A 16-character local identifier of a UOW.

ALL
The default. If you do not specify a UOW identifier, information about all UOWs in the system is shown.

Sample screen

```
IN UOW
STATUS:  RESULTS - OVERTYPE TO MODIFY
 Uow(AB8FFF37584B6601) Inf Act Tra(CSSY) Tas(0000005)
    Age(00004090)                         Use(BELL    )
 Uow(AB8FFF3758786C01) Inf Act Tra(CSSY) Tas(0000006)
    Age(00004090)                         Use(BELL    )
 Uow(AB8FFF376D9B6601) Inf Act Tra(CSTP) Tas(0000008)
    Age(00004090)                         Use(BELL    )
 Uow(AB900BD865417C04) Inf Act Tra(CSNE) Tas(0000018)
    Age(00000700)                         Use(BELL    )
 Uow(AB900BE2010AC401) Inf Act Tra(CEMT) Tas(0000043)
    Age(00000689) Ter(S21D) Netn(IGBS21D ) Use(BELL    )
```

Figure 114. CEMT INQUIRE UOW screen

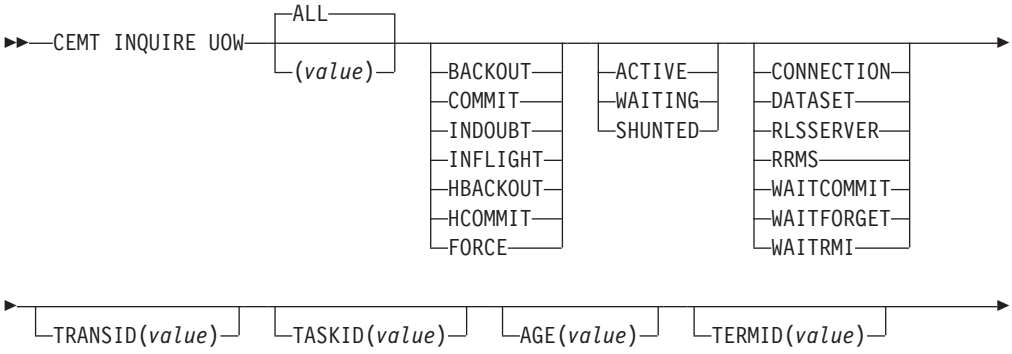
Note: When a value does not apply, or is negative (the value begins with No), the fields on the screen are blank. To modify these fields, locate them by tabbing (they appear in the same sequence as in the expanded format), and overtype with input valid for that field. You might find it more convenient to use the expanded format when setting one of these values.

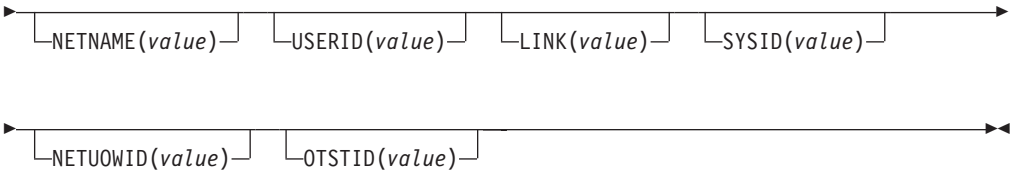
If you place the cursor against a specific entry in the list and press ENTER, CICS displays an expanded format, as shown in Figure 115.

```
IN UOW
RESULT - OVERTYPE TO MODIFY
  Uow(AB8FFF37584B6601)
  Uowstate( Inflight )
  Waitstate(Active)
  Transid(CSSY)
  Taskid(0000005)
  Age(00004233)
  Termid()
  Netname()
  Userid(BELL)
  Waitcause()
  Link()
  Sysid()
  Netuowid(..GBIBMIYA.CICSHT61........)
  Otstid()
```

Figure 115. The expanded display of an individual entry

CEMT INQUIRE UOW





Displayed fields

AGE (*value*)
Displays the age of the UOW in seconds. This is the number of seconds since the UOW entered its current state, or since the start of the UOW.

LINK(*value*)
Displays the 8-character value that, for a WAITCAUSE value of CONNECTION, is the netname of the remote system that caused the UOW to wait or be shunted. For other WAITCAUSE values, LINK displays blanks.

NETNAME (*value*)
Displays the 8-character network name of the terminal from which the UOW was started.

If the UOW was started from an ISC or MRO session, NETNAME displays the network name of the remote region.

If the UOW was not started from a terminal, nor from an ISC or MRO session, NETNAME displays blanks.

NETUOWID (*value*)
Displays a 27-character string that contains the LU6.2 name for the UOW within this network; that is, the network-wide identifier of the UOW.

Nondisplayable characters appear as periods. You can use PF2 on the expanded panel: to see the value in hexadecimal.

You can assemble information about a distributed UOW by matching the network-wide UOW identifier against the network-wide identifiers of local UOWs on other systems.

OTSTID(*value*)
Displays the first 128 bytes of the transaction identifier (TID) of the OTS transaction of which the UOW is a part. If the OTS name has fewer than 128 bytes, it is padded on the right with binary zeros.

SYSID (*value*)
Displays the 4-character sysid that, for a WAITCAUSE value of CONNECTION, is the sysid of the connection that caused the UOW to wait ir be shunted. If the connection has been discarded, and for other WAITCAUSE values, SYSID returns blanks.

TASKID (*value*)
Displays the task number originally associated with this UOW. If the UOW is shunted, the task terminates. In this case, the number may have been reused by another task.

TERMID (*value*)
Displays the 4-character ID of the terminal or session from which this UOW was started. This is the principal facility for the task. If the transaction is the

mirror transaction, CSMI, it is the session. For OUWs that are part of an OTS transaction, it will be the session used by the request stream that attached the task.

TRANSID (*value*)
Displays the 4-character ID of the transaction that started this UOW.

UOW(*value*)
Indicates that this panel relates to a UOW inquiry and displays a 16-character UOW local identifier.

UOWSTATE
Displays the state of the UOW. The values are as follows:

BACKOUT
This UOW is being backed out, or has failed to back out one or more of the recoverable resources involved in the UOW.

COMMIT
A decision to commit the UOW has been made, but the UOW is waiting or has been shunted. This might be because the decision has not yet been communicated to all participants in the syncpoint, or because a failure has occurred during commit processing.

INDOUBT
This UOW is in the indoubt state.

INFLIGHT
The UOW is running normally.

HBACKOUT
The UOW has been forcibly backed out. A forced decision is taken when a UOW is unable to wait for indoubt resolution; for example, the transaction was defined as WAIT(NO), or was backed out with a CEMT SET UOW command.

HCOMMIT
The UOW has been forcibly committed.

Note: You can reset this value by overtyping it with:

COMMIT
Attempt to force syncpoint commit processing, as specified for this UOW.

BACKOUT
Attempt to force syncpoint backout processing, as specified for this UOW.

FORCE
Attempt to force the UOW to backout or commit, as specified on the ACTION option of the TRANSACTION resource definition.

All these values are valid only for UOWs that have been shunted indoubt. For information about the indoubt attributes of TRANSACTION resource definition, see the *CICS Resource Definition Guide*.

USERID(*value*)
Displays the 8-character user ID with which this transaction was started.

WAITCAUSE
Displays, if the UOW is waiting or shunted, the type of resource that caused it to be so.

Note: In the case of a wait, it is the UOW that is waiting, not the task.

The values are as follows:

CONNECTION
This UOW is waiting or was shunted because of the failure of a session to the coordinator of the UOW during the indoubt period. NETNAME and SYSID contain the netname and system name of the failed link.

DATASET
This UOW is waiting or was shunted because of the failure of one or more data sets. Use the INQUIRE UOWDSNFAIL command to identify the data sets involved and the reasons why they have caused the UOW to fail.

RLSSERVER
This UOW is waiting or was shunted for the OTS coordinator because of the failure of an RLS server.

RRMS
This UOW is waiting or has been shunted because communication has been lost with RRS/MVS.

WAITCOMMIT
This UOW is waiting or was shunted because a failure occurred during commit processing.

WAITFORGET
This UOW is waiting for FORGET from participants in the syncpoint. Use the INQUIRE UOWLINK command to obtain the netnames and sysids of the participants.

WAITRMI
This UOW is waiting for FORGET from the RMI. Use the **INQUIRE UOWLINK** command to obtain the entry name and qualifier of the task-related user exit.

WAITSTATE
Displays whether syncpoint processing of the UOW has been deferred. The values are as follows:

ACTIVE
The UOW is running normally, or is being unshunted.

WAITING
Syncpoint processing has completed on this system, but not on all systems involved in the distributed UOW. Values of WAITFORGET or WAITRMI, and BACKOUT or COMMIT indicate how the UOW was resolved on this system.

SHUNTED
Syncpoint processing of the UOW has been deferred. SHUNTED further indicates that the task, terminal and program storage have been released, and locks have been retained.

CEMT INQUIRE UOWDSNFAIL

Obtain information about failed units of work that have updated CICS file-control-managed data sets.

Description

The **INQUIRE UOWDSNFAIL** command returns information about unit of work (UOW) failures that are associated with specific data sets. If you enter the basic command, **CEMT INQUIRE UOWDSNFAIL**, without any attributes, all failed UOWs for all data sets are displayed.

If a UOW that had suffered a failure is in the process of being retried when the **INQUIRE UOWDSNFAIL** command is issued, such a UOW is not listed (it is not currently shunted or failed). If the retry is successful, the UOW no longer has a failure with respect to that particular data set. If the retry is not successful, the UOW will be shunted again, and INQUIRE UOWDSNFAIL will display it as failed with respect to the data set.

Alternatively, if you are interested only in those failed UOWs for specific data sets, you can use the DSNAME parameter as a filter by specifying either a specific or generic data set name. You can also use other attributes as filters, for example, CONNECTION for UOWs that failed indoubt, or DATASET for UOWs that failed during backout. See the list of reasons for UOW failures, and the associated causes, that you can use to filter the results of INQUIRE UOWDSNFAIL commands.

Input

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT INQUIRE UOWDSNFAIL` (or suitable abbreviations for the keywords). The resulting display lists all the failed UOWs, if any. Otherwise, CICS returns the NOT FOUND error condition.
- Type `CEMT INQUIRE UOWDSNFAIL` (or suitable abbreviations for the keywords), followed by the attributes that are necessary to limit the range of information that you require. For example, if you enter `cemt i uowd ds(applb.*)`, the resulting display shows the details of only those failed UOWs that are associated with data sets with a high-level qualifier of APPLB.

Sample screen

```
INQ UOWDSN
STATUS:  RESULTS
 Dsn(RLS.ACCOUNTS.ESDS.DBASE1                  ) Dat Del
    Uow(AA6DB080C40CEE01)                       Rls
 Dsn(RLS.ACCOUNTS.ESDS.DBASE1                  ) Dat Ind
    Uow(AA6DB08AC66B4000)                       Rls
```

Figure 116. CEMT INQUIRE UOWDSNFAIL screen

Sample screen

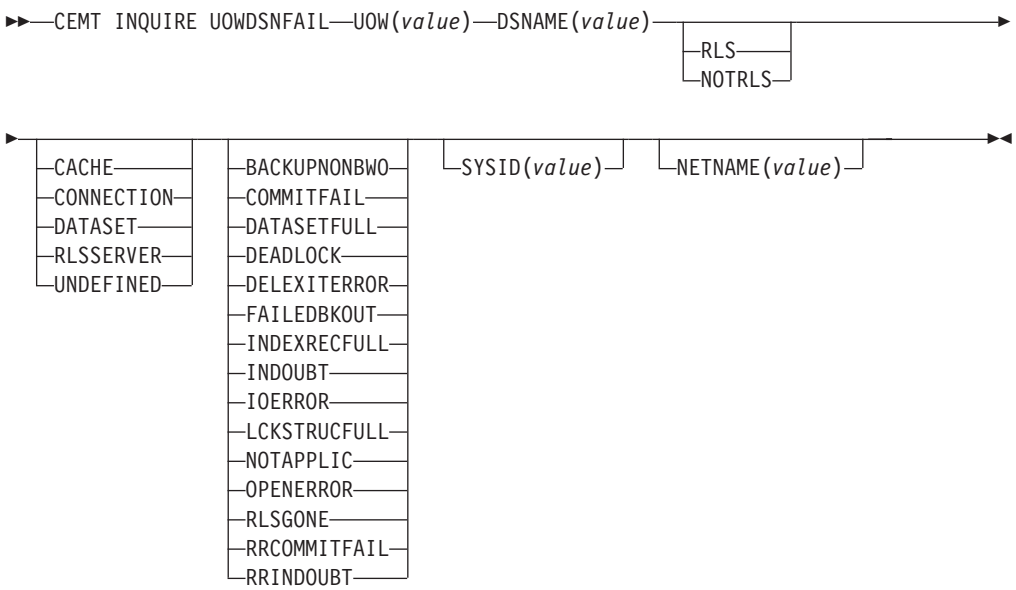
Note: When a value does not apply, or is negative (the value begins with No), the fields on the screen are blank. To modify these fields, locate them by tabbing (they appear in the same sequence as in the expanded format), and overtype with input valid for that field. You might find it more convenient to use the expanded format when setting one of these values.

If you place the cursor against a specific entry in the list and press ENTER, CICS displays an expanded format, as shown in Figure 117 on page 374.

```
INQ UOWDSN
RESULT
  Uowdsnfail
  Dsname(RLS.ACCOUNTS.ESDS.DBASE1)
  Cause(Dataset)
  Reason(Delexiterror)
  Uow(AA6DB080C40CEE01)
  Sysid()
  Netname()
  Rlsaccess(Rls)
```

Figure 117. The expanded display of an individual entry

CEMT INQUIRE UOWDSNFAIL



Displayed fields

CAUSE
Displays which failed component has caused the UOW to have retained locks for this data set. The values are as follows:

CACHE
A cache structure, or connection to it, has failed.

CONNECTION
There is an intersystem connection error, which has caused the UOW to fail while indoubt. The name of the system to which connectivity was lost is returned on the SYSID parameter and its netname is returned on the NETNAME parameter. CICS returns additional information in the REASON parameter about the CONNECTION failure.

DATASET
The backout of a UOW has failed for this data set. The reason for the data set failure is returned in the REASON parameter.

RLSSERVER
The SMSVSAM server has failed. The reason for the data set failure is returned in the REASON parameter.

UNDEFINED
The UOW is probably in-flight following an emergency restart.

DSNAME (*value*)
Displays the 44-character data set name of a data set that has retained locks for this UOW.

NETNAME (*value*)
Displays the 8-character netname (when the cause is CONNECTION) of the remote system to which connectivity has been lost.

REASON
Displays, when the cause is RLSSERVER, CONNECTION, or DATASET, the specific reason for the error against this data set. The values are as follows:

BACKUPNONBWO
A non-BWO backup was in progress while the UOW was being backed out. When the backup completes, CICS automatically retries the UOW.

COMMITFAIL
An error occurred at some point when the locks were in the process of being released. This is an error that can normally be resolved by recycling the server (which should happen automatically). The locks were acquired as a result of recoverable requests having been issued against the data set.

DATASETFULL
No space is available on the direct access device for adding records to a data set. You need to reallocate the data set with more space. You can then retry the backout using SET DSNAME RETRY.

See Moving recoverable data sets that have retained locksin the *CICS Recovery and Restart Guide*.

DEADLOCK (*non-rls data sets only*)
A deadlock was detected. This may go away if the UOW is retried.

DELEXITERROR
Backout of a write to an ESDS failed because a logical delete global user exit program was not enabled, or a logical delete global user exit program decided not to execute the logical delete.

FAILEDBKOUT
This occurs as a result of a severe error being identified, and is possibly an error in either CICS or VSAM. The problem may go away if the UOW is retried. Note that CICS performs some first-failure data capture (FFDC) at the point where the error is first detected.

INDEXRECFULL
A larger alternate index record size needs to be defined for the data set. See Moving recoverable data sets that have retained locksin the *CICS Recovery and Restart Guide*.

This error can also occur when a unique alternate index key, for a non-RLS data set, has been reused and CICS is now backing out the request which had removed that key value.

INDOUBT
The unit of work had issued recoverable requests against the data set, and has now failed indoubt. The connection to the coordinating system needs to be reestablished.

IOERROR
A hard I/O error occurred. To correct this error, restore a full backup copy of the data set and perform forward recovery. If you use CICS VSAM Recovery as your forward recovery utility, the backout is automatically retried for an RLS data set. For a non-RLS data set, use the RETRY option of the SET DSNAME command to drive the backout retry.

LCKSTRUCFULL
An attempt to acquire a lock during backout of an update to this data set failed because the RLS lock structure was full. You must allocate a larger lock structure in an available coupling facility and rebuild the existing lock structure into it, then use the SET DSNAME (...) RETRY command to drive the backout retry.

NOTAPPLIC
The value for CAUSE is not CONNECTION, RLSSERVER, or DATASET.

OPENERROR
Error on opening the file for backout. A console message notifies you of the reason for the open error. One likely reason could be that the data set was quiesced.

RLSGONE
An error occurred when backing out the UOW because the RLS server is unavailable. This might also be the reason why the UOW went into backout originally. This is an error that can be resolved by recycling the server (which should happen automatically). Generally, when the server recovers, the UOWs are retried automatically. In very exceptional circumstances, it may be necessary to issue a **SET DSNAME(...) RETRY** command to retry UOWs that were not retried when the server returned.

RRCOMMITFAIL
An error occurred while RLS locks for the unit of work were being released. For this data set, the locks being released were all repeatable read locks so, if the failure was due to the RLS server being unavailable, the locks will have been released. If the failure was due to some other error from the SMSVSAM server, the locks may still be held.

RRINDOUBT
The unit of work had issued repeatable read requests against the data set, and has now failed indoubt. The locks will have been released, so this failure does not prevent you from running a batch job against the data set. However, if you want to open the data set in non-RLS mode from CICS, you need to resolve the indoubt failure before you can define the file as having RLSACCESS(NO). If the unit of work has updated any other data sets, or any other resources, you should try to resolve the indoubt correctly. If the unit of work has only performed repeatable reads against VSAM data sets and has made no updates to other resources, it is safe to force the unit of work using the SET DSNAME or SET UOW commands.

Each Reason corresponds to only one Cause value, as shown in the following table:

Cause	Reason
CACHE	NOTAPPLIC
CONNECTION	INDOUBT
CONNECTION	RRINDOUBT
DATASET	BACKUPNONBWO
DATASET	DELEXITERROR
DATASET	DATASETFULL
DATASET	DEADLOCK
DATASET	FAILEDBKOUT
DATASET	INDEXRECFULL
DATASET	IOERROR

Cause	Reason
DATASET	LCKSTRUCFULL
DATASET	OPENERROR
RLSSERVER	COMMITFAIL
RLSSERVER	RLSGONE
RLSSERVER	RRCOMMITFAIL
UNDEFINED	Not applicable

RLSACCESS
Displays whether the data set was last opened in this CICS region in RLS or non-RLS mode. The values are as follows:

RLS The last open in this CICS region was in RLS mode.

NOTRLS
The last open in this CICS region was in non-RLS mode.

SYSID(*value***)**
Displays the 4-character SYSID (when the cause is CONNECTION) of the remote system to which connectivity has been lost.

UOW (*value***)**
Displays the 16-character local identifier of the associated unit of work.

UOWDSNFAIL
Indicates that this panel relates to an inquiry about the cause of a UOW failure.

CEMT INQUIRE UOWENQ

Retrieve information about enqueues held or waited on by a unit of work (UOW), or about UOWs holding or waiting on a specified enqueue.

In the CICS Explorer, the Unit of work enqueues operations view provides a functional equivalent to this command.

Description

The **INQUIRE UOWENQ** command returns information about the enqueues held by a UOW. Enqueues are used by CICS to lock recoverable resources such as file records or queues to the UOW that is updating them. User enqueues obtained by the **EXEC CICS ENQ** command are also returned.

You can use the **INQUIRE UOWENQ** command to indicate which records and queues would be affected if the UOW were forced.

The transaction also returns information about UOWs that are waiting on the enqueues. You can use this to diagnose enqueue deadlocks between tasks that want to update the same resources.

Enqueues are typically held in active state, which means that other tasks are allowed to wait for the enqueue. However, if the UOW that owns the enqueue suffers an indoubt failure, the enqueue is usually converted to the retained state until the indoubt failure can be resolved. The **INQUIRE UOWENQ** command also retrieves information about these enqueues and can be used to identify which records and queues would be affected if the UOW were forced.

Note:

1. Both UOW-lifetime and task-lifetime enqueues are returned by the **INQUIRE UOWENQ** command. (For an explanation of UOW- and task-lifetime enqueues, see the MAXLIFETIME option of the **EXEC CICS ENQ** command.)
2. On an indoubt failure, user enqueues are released, *unless* the EXEC CICS ENQ command specified MAXLIFETIME(TASK) and it is not the end-of-task syncpoint that suffers the failure.

Input

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT INQUIRE UOWENQ` (or suitable abbreviations for the keywords). The resulting display lists all enqueues for all the current UOWs.
- Type `CEMT INQUIRE UOWENQ` (or suitable abbreviations for the keywords), followed by the attributes that are necessary to limit the range of information that you require. For example, if you enter `cemt i uowe d`, the resulting display shows the details only of enqueues on records in VSAM data sets open in non-RLS mode or CICS-maintained data tables.

Sample screen

```
INQ UOWE
STATUS:  RESULTS
 Uow(AC143001D4A72601) Tra(CECI) Tas(0000038) Act Exe Own
    Res(ACCOUNTS.DBASE1           ) Rle(015) Enq(00000000)
 Uow(AC143001D4A72601) Tra(CECI) Tas(0000038) Act Add Own
    Res(g...                      ) Rle(004) Enq(00000000)
```

Figure 118. CEMT INQUIRE UOWENQ screen

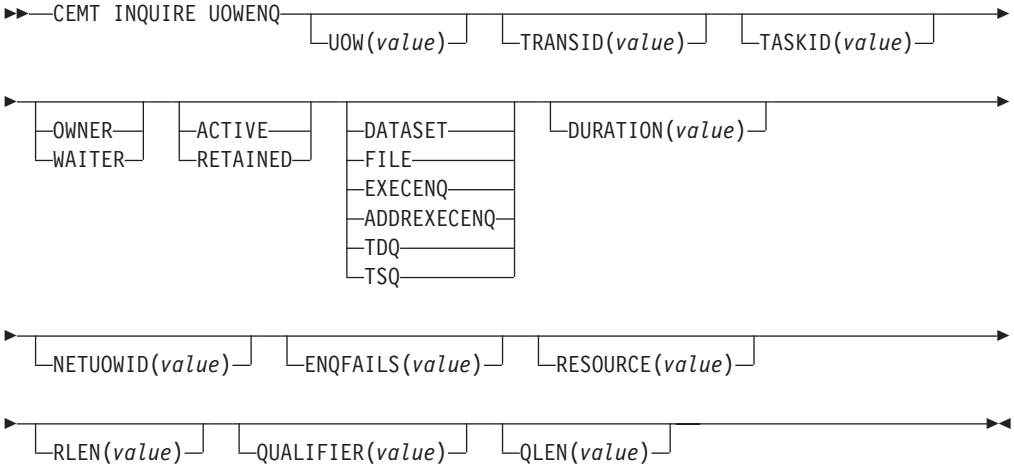
Note: When a value does not apply, or is negative (the value begins with No), the fields on the screen are blank. To modify these fields, locate them by tabbing (they appear in the same sequence as in the expanded format), and overtype with input valid for that field. You might find it more convenient to use the expanded format when setting one of these values.

If you place the cursor against a specific entry in the list and press ENTER, CICS displays an expanded format, as shown in Figure 119.

```
INQ UOWE
RESULT
  Uowenq
  Uow(AC143001D4A72601)
  Transid(CECI)
  Taskid(0000038)
  State(Active)
  Type(Execenq)
  Relation(Owner)
  Resource(ACCOUNTS.DBASE1)
  Rlen(015)
  Enqfails(00000000)
  Netuowid(..GBIBMIYA.IYA2T5C2...Mx...)
  Qualifier()
  Qlen(000)
```

Figure 119. The expanded display of an individual entry

CEMT INQUIRE UOWENQ



Displayed fields

DURATION(*value*)
Displays the elapsed time (in seconds) since the enqueue entered its current state of OWNER, WAITER, or RETAINED.

ENQFAILS (*value*)
Displays, for enqueues held in retained state, the number of failed enqueue attempts for this resource since the enqueue was last acquired. This indicates how many UOWs have received LOCKED responses due to this enqueue. The ENQFAILS option helps you identify which UOWs are causing bottlenecks.

For enqueues held in action state, ENQFAILS displays zero.

ENQSCOPE (*value*)
If the enqueue has sysplex scope, ENQSCOPE displays the 4-character name that was used to qualify the sysplex-wide ENQUEUE request issued by this CICS region. If it has region scope, ENQSCOPE displays blanks.

All CICS systems with the same ENQSCOPE value share the same sysplex Enqueue namespace.

NETUOWID(*value*)
Displays a 27-character string that contains the network-wide LU6.2 name of the UOW that owns, or is waiting for, the enqueue.

Note: Nondisplayable characters appear as periods. You can use PF2 on the expanded panel: to see the value in hexadecimal.

QLEN (*value*)
Displays the length of the data, in the range 0 through 255, displayed in the QUALIFIER field.

QUALIFIER (*value*)
Displays a 1- through 255-character qualifier (for example, record id) that further identifies the resource associated with the enqueue lock. The data displayed depends on the type of enqueue, as shown in Table 4 on page 381. If no QUALIFIER data applies (that is for EXECENQ, ADDREXECENQ, or TSQ), a value of zero is displayed.

RELATION
Displays whether the data is associated with the owner of the enqueue or with a task waiting for the enqueue. The values are as follows:

OWNER
The (*value*), NETUOWID, TASKID, and TRANSID are those of the owner of the enqueue.

WAITER
The (*value*), NETUOWID, TASKID, and TRANSID are those of a waiter for the enqueue.

RESOURCE (*value*)
Displays a 1- through 255-character string that contains the name of the resource associated with the enqueue lock. The data displayed depends on the type of enqueue, as shown in Table 4 on page 381.

Note: Nondisplayable characters appear as periods. You can use PF2 on the expanded panel: to see the value in hexadecimal.

RLEN (*value*)
Displays the length of the data, in the range 0 through 255, displayed in the RESOURCE field.

STATE
Displays the state that the enqueue is held in. The values are as follows:

ACTIVE
The enqueue is held in active state.

RETAINED
The enqueue is held in retained state. Its owning UOW has been shunted, or is in the process of being shunted.

TASKID (*value*)
Displays the number of the task associated with the UOW. If the UOW is shunted, it is the task number associated with the UOW before it was shunted.

TRANSID (*value*)
Displays the 4-character identifier of the transaction associated with the UOW. If the UOW is shunted, it is the identifier of the transaction associated with the UOW before it was shunted.

TYPE
Displays the type of resource being enqueued upon. The values are as follows:

DATASET
The resource is a record in a VSAM data set (or a CICS-maintained data table). RESOURCE contains the name of the data set and QUALIFIER contains the record identifier.

FILE The resource is a record in either a BDAM file or a user-maintained data table. RESOURCE contains the name of the file and QUALIFIER contains the record identifier.

When the file is a BDAM file, the record identifier is prefixed by the BDAM block identifier. If this combination exceeds 255 characters, truncation occurs.

EXECENQ
The resource is associated with an EXEC CICS ENQ request. RESOURCE contains the enqueue argument passed on the request.

ADDREXECENQ
The resource is associated with an EXEC CICS ENQ request. RESOURCE contains the address enqueue argument passed on the request (that is, the LENGTH parameter was omitted on the request).

TDQ The resource is a logically-recoverable transient data queue. RESOURCE contains the name of the queue. QUALIFIER contains either the string FROMQ or TOQ, indicating whether a read or write lock is held for the queue.

A READQ TD request acquires the FROMQ lock, whereas a WRITEQ TD request acquires the TOQ lock associated with the queue. A DELETEQ TD request acquires both the TOQ and the FROMQ locks.

TSQ The resource is a recoverable temporary storage queue. RESOURCE contains the name of the queue.

The data displayed in the RESOURCE and QUALIFIER fields depends on the resource type, as shown in Table 4.

Table 4. Data displayed in RESOURCE and QUALIFIER

Type	Resource	Qualifier
DATASET	Data set name	Record identifier
EXECENQ	EXEC enqueue argument	None
ADDREXECENQ	Address of EXEC enqueue argument	None
FILE	File name	Record identifier
TDQ	TD queue name	FROMQ or TOQ
TSQ	TS queue name	None

UOW(*value***)**
Displays the 16-character local identifier of the associated unit of work.

CEMT INQUIRE UOWLINK

Retrieve information about connections involved in units of work (UOW).

Description

The **INQUIRE UOWLINK** command returns information about a connection involved in a unit of work. The connection can be to a remote system or to a task-related user exit. If it is to a remote system, **INQUIRE UOWLINK** displays the netname of the connection, its SYSID, and whether it is the coordinator or subordinate. If it is to a task-related user exit, **INQUIRE UOWLINK** displays the entry name and qualifier of the exit.

Input

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT INQUIRE UOWLINK` (or suitable abbreviations for the keywords). The resulting display lists all the current UOW links.
- Type `CEMT INQUIRE UOWLINK` (or suitable abbreviations for the keywords), followed by the attributes that are necessary to limit the range of information that you require. For example, if you enter `cemt i uowl rmi`, the resulting display shows details only of UOW links to external resource managers.

(*value***)**
An 8-byte token that identifies the UOW link for which data is returned.

ALL
The default.

Sample screen

```
I UOWL
STATUS:  RESULTS - OVERTYPE TO MODIFY
 Uowl(01840009) Uow(AC142A977E6F3204) Con Lin(IYLX1   )
    Coo Appc War Sys(ISC1)               Net(..GBIBMIYA.IYCWTC13..p...
 Uowl(01C40006) Uow(AC142A977E6F3204) Con Lin(IYLX3   )
    Sub Irc  War Sys(MRO3)               Net(..GBIBMIYA.IYCWTC13..p...
```

Figure 120. CEMT INQUIRE UOWLINK screen

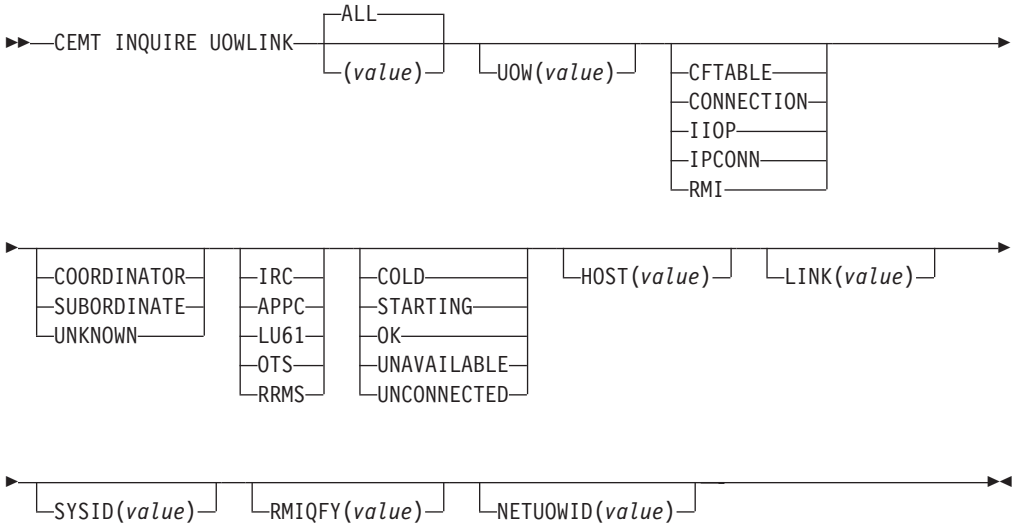
Note: When a value does not apply, or is negative (the value begins with No), the fields on the screen are blank. To modify these fields, locate them by tabbing (they appear in the same sequence as in the expanded format), and overtype with input valid for that field. You might find it more convenient to use the expanded format when setting one of these values.

If you place the cursor against a specific entry in the list and press Enter, CICS displays an expanded format, as shown in Figure 121.

```
 I UOWL
 RESULT - OVERTYPE TO MODIFY
   Uowlink(01840009)
   Uow(AC142A977E6F3204)
   Type(Connection)
   Host( )
   Link(IYLX1)
   Action(             )
   Role(Coordinator)
   Protocol( )
   Resyncstatus( )
   Sysid( )
   Rmiqfy( )
   Netuowid( )
   Urid
  Xid
```

Figure 121. The expanded display of an individual entry

CEMT INQUIRE UOWLINK



Displayed fields

HOST(*value***)**
For TYPE values IIOP and IPCONN, displays a 255-character area into which the TCP/IP host name is returned, or a string containing the dotted decimal TCP/IP address, used to refer to the participant in the OTS transaction. Strings with fewer than 255 characters are padded with blanks. The padding is useful for identifying the participant, especially when problems occur. The UOW and the participant might belong to the same CorbaServer.

For other TYPE values, this value displays blanks.

LINK (*value***)**
The value returned depends on the TYPE returned:

CFTABLE
The 8-character name of the coupling facility data table pool.

CONNECTION
The 8-character netname of the remote system.

IIOP
Blanks. The equivalent IIOP information is returned by the HOST attribute.

IPCONN
The 8-character APPLID of the remote system.

RMI
The entry name of the task-related user exit.

Displays, for a TYPE value of CONNECTION, the 8-character netname of the remote system. For a TYPE value of RMI, LINK displays the entry name of the task-related user exit. For a TYPE value of CFTABLE, Link displays the 8–character name of the coupling facility data table pool. For a type value of IIOP, LINK displays blanks; the equivalent information for IIOP unit-of-work links are displayed by the HOST attribute.

NETUOWID (*value***)**
Displays a 27-character string containing the network-wide LU6.2 name for the UOW.

Nondisplayable characters appear as periods. You can use PF2 on the expanded panel to see the value in hexadecimal.

PROTOCOL
Displays the communication protocol used by the connection. The values are as follows:
IRC Interregion Communication. This connection is an MRO connection.
APPC
Advanced Program to Program Communication.
LU61 LUTYPE 6.1.
IPIC IPIC connection.
OTS Object Transaction Services for Java programs.
RRMS
Indicates that the UOW is coordinated by RRS/MVS.

RESYNCSTATUS
Displays the resynchronization status of the connection. The values are as follows:
COLD
The connection has been cold started by the partner system. The

partner can no longer coordinate any in-doubts for this system; nor can this system pass to the partner any decisions remembered for it.

STARTING
The connection is being acquired, but the exchange lognames process has not yet completed.

OK The connection is operating normally. If the partner system has failed, the partner has been restarted and the connection can resynchronize the associated UOW.

UNAVAILABLE
The connection is not currently acquired.

UNCONNECTED
No associated connection.

RMIQFY (*value*)
Displays, for a connection type of RMI, the 8-character entry qualifier of the task-related user exit. For a connection type of CONNECTION, IPCONN, CFTABLE, or IIOP, RMIQFY displays blanks.

ROLE
Displays the role of the connection. The values are as follows:

COORDINATOR
This connection is to the sync point coordinator for the UOW.

SUBORDINATE
This connection is to a sync point subordinate for the UOW.

UNKNOWN
The sync point role of this connection cannot be determined.

SYSID (*value*)
Displays, for a TYPE value of CONNECTION, the 4-character SYSID of the connection. If the connection has been discarded, or the connection type is RMI, IPCONN, CFTABLE or IIOP, or the PROTOCOL field is showing RRMS, SYSID displays blanks.

TYPE
Displays the type of connection. The values are as follows:

CFTABLE
A connection to a CFDT server.

CONNECTION
A connection defined by a CONNECTION resource definition.

IIOP An OTS link.

IPCONN
A connection defined by an IPCONN resource definition.

RMI A connection to an external resource manager using the resource manager interface (RMI).

UOWLINK(*value*)
Indicates that this panel relates to an inquiry about a UOW link and displays the 8-character UOW link identifier.

UOW (*value*)
Displays the 16-character local identifier of the associated unit of work.

URID (*value*)
If the PROTOCOL field displays RRMS, this field contains the 32-byte hexadecimal representation of the RRMS unit of recovery identifier. For other values of the PROTOCOL field, including OTS, URID displays blanks.

XID(*data-area*)
Returns the 64-character global transaction identifier, if present.

CEMT INQUIRE URIMAP

Retrieve information about URIMAP resource definitions.

In the CICS Explorer, the URI maps operations view provides a functional equivalent to this command.

Description

The USAGE attribute of a URIMAP definition determines which other attributes are specified in that URIMAP definition and sometimes determines the meaning of a particular attribute.

The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key to clear the screen. You can start this transaction in two ways:

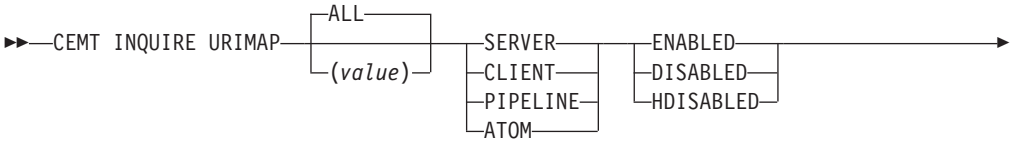
- Type `CEMT INQUIRE URIMAP` (or suitable abbreviations for the keywords). The resulting display lists the current status.
- Type `CEMT INQUIRE URIMAP` (or suitable abbreviations for the keywords), followed by the attributes that are necessary to limit the range of information that you require. For example, if you enter `cemt i ur tcpipservice`, the resulting display shows the details of only those URIMAP resource definitions that apply to the selected TCPIPSERVICE definition (which defines a port).

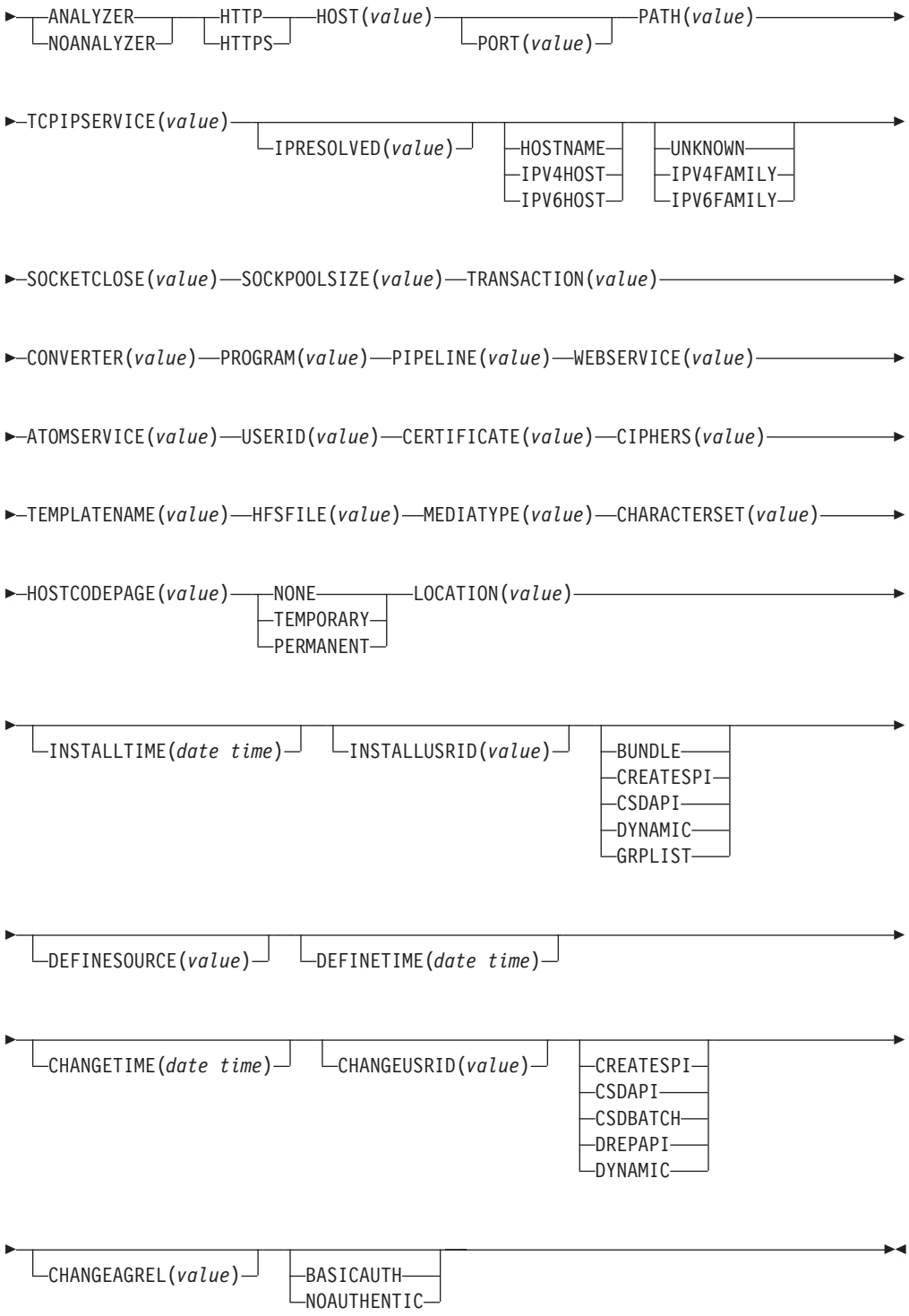
You can tab to the highlighted or blank fields and overtype them with the required values.

ALL
The default.

(*value*)
The name of the URIMAP definition for which information is requested.

CEMT INQUIRE URIMAP





Displayed fields

ANALYZERSTAT(*value***)**
Displays the analyzer program setting for the URIMAP definition. The possible values are as follows:

ANALYZER
The analyzer associated with the TCPIPSERVICE definition is to be used in processing the HTTP request.

NOANALYZER
The analyzer program is not to be used.

ATOMSERVICE(*value***)**
Displays the 1- to 8-character name of an ATOMSERVICE resource definition for an Atom feed. The ATOMSERVICE resource definition defines an Atom service, feed, collection, or category document, and identifies the Atom configuration file, CICS resource or application program, and XML binding that are used to supply the data for the feed.

AUTHENTICATE(*value***)**
Displays whether to provide authentication information to a web services provider. The possible values are as follows:

BASICAUTH
The web services provider requires HTTP basic authentication. You can supply credentials to the web services requester (a user ID and password) to the global user exit, XWBAUTH, which, if enabled, sends the credentials to the web services provider.

NOAUTHENTIC
The web services provider does not require authentication.

CERTIFICATE(*value***)**
Displays the label of the certificate that is to be used as the SSL client certificate for the HTTP request by CICS as an HTTP client. The certificate label can be up to 32 characters.

CHANGEAGENT(*value***)**
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

DYNAMIC
The resource definition was last changed by a PIPELINE scan.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The

format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value*)
Displays the 8-character user ID that ran the change agent.

CHARACTERSET(*value*)
Displays the name of the character set for the static response. The name of the character set can be up to 40 characters.

CIPHERS(*value*)
Displays the cipher code that is to be used for the HTTP request by CICS as an HTTP client,.The cipher code can be up to 48 characters.

CONVERTER(*value*)
Displays the 8-character name of a converter program that performs conversion or other processing on the content of the request.

DEFINESOURCE(*value*)
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time*)
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

ENABLESTATUS(*value*)
Displays the status of the URIMAP definition. The possible values are as follows:

ENABLED
The URIMAP definition can be accessed.

DISABLED
The URIMAP definition cannot be accessed. A URIMAP definition with this status can be deleted.

HDISABLED
The URIMAP definition cannot be accessed because the virtual host of which it forms a part has been disabled. Use CEMT SET HOST to reenable all the URIMAP definitions in the virtual host. A URIMAP definition with this status cannot be deleted.

HFSFILE(*value*)
Displays the fully qualified name of a UNIX System Services file that will form the static response. The fully qualified file name can be up to 256 characters.

HOST(*value*)
Displays the host name, IPv4, or IPv6 address of the URL. All IPv4 addresses are displayed as native IPv4 dotted decimal addresses; for example, `1.2.3.4`, regardless of the type of address format used. If you are using a URIMAP definition relating to CICS as an HTTP client, USAGE(CLIENT), and a native IPv4 address or host name is used, the address or host name and port number are displayed; for example, `1.2.3.4:80` or `hostname.com:443`. If you specify a hostname that is greater than 110 characters in length, port information is not displayed in the HOST option. Use the PORT option to view the port number.

HOSTCODEPAGE(*value***)**
Displays the IBM codepage (EBCDIC) in which the text document that will form the static response is encoded. This value can be up to 10 characters.

HOSTTYPE(*value***)**
Displays the address format of the HOST option. The HOSTTYPE option is available using an expanded query only. The possible values are as follows:

HOSTNAME
The HOST option contains a character host name.

IPV4HOST
The address is an IPv4 address.

IPV6HOST
The address is an IPv6 address.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

BUNDLE
The resource was installed by a bundle deployment.

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

DYNAMIC
The resource was installed by using a PIPELINE scan.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

IPFAMILY(*value***)**
Displays the address format of the IPRESOLVED option. The IPFAMILY option is available using an expanded query only. Filtering interacts with the IPFAMILY option when you filter using wildcard characters. For example, if the value in IPFAMILY is IPV6FAMILY, you must use the colon character to retrieve an IP address. The possible values are as follows:

IPV4FAMILY
The address is an IPv4 address.

IPV6FAMILY
The address is an IPv6 address.

UNKNOWN
The IPRESOLVED option is not yet in use, or the address cannot be resolved.

IPRESOLVED(*value*)
Displays the IPv4 or IPv6 address of the host. The IPRESOLVED option is available using an expanded query only. This option is for all types except USAGE(SERVER).

LOCATION(*value*)
Displays a URL to which matching HTTP requests from Web clients are redirected. The URL can be up to 255 characters. Redirection is activated by the setting specified by the REDIRECTTYPE option.

MEDIATYPE(*value*)
Displays a description of the data content of the static response. The media type can be up to 40 characters.

PATH(*value*)
Displays the path component of the URL. The path component can be up to 255 characters.

PORT(*value*)
Displays the numeric port number value used by USAGE(CLIENT) for the connection to the server, in the range 1 - 65535. The port number is also displayed in the HOST option if HOST contains a native IPv4 address or a host name. For USAGE(CLIENT), the PORT attribute always contains the port number that is being used for the communication, even if PORT(NO) is specified on the URIMAP at define time. For USAGE(ATOM), USAGE(SERVER), or USAGE(PIPELINE), the PORT option displays ().

PROGRAM(*value*)
Displays the 8-character name of the application program that composes a response.

REDIRECTTYPE(*value*)
Displays the type of redirection for requests that match this URIMAP definition. The URL for redirection is specified by the Location option. The possible values are as follows:

NONE
Requests are not redirected. Any URL specified by the LOCATION option is ignored.

TEMPORARY
Requests are redirected on a temporary basis. The status code used for the response is 302 (Found).

PERMANENT
Requests are redirected permanently. The status code used for the response is 301 (Moved Permanently).

SCHEME(*value*)
Displays the scheme component of the URL. The possible values are as follows:

HTTP HTTP without Secure Sockets Layer (SSL)

HTTPS
HTTP with Secure Sockets Layer (SSL)

SOCKETCLOSE(*value*)
Displays the maximum length of time in seconds that CICS keeps a client HTTP connection open for reuse after the CICS application has finished using it. If the value is 0, CICS does not keep connections open for reuse. A value is

only displayed when the URIMAP definition has a usage type of CLIENT. For other usage types, SOCKETCLOSE displays ().

SOCKPOOLSIZE(*value*)
Displays the number of client HTTP connections that CICS is currently holding in a pool in a dormant state. The connections can be reused by any CICS application that connects as a Web client to the same host and port. A value is only displayed when the URIMAP definition has a usage type of CLIENT. For other usage types, SOCKPOOLSIZE displays ().

TCPIPSERVICE(*value*)
Displays the 8-character name of the TCPIPSERVICE definition that specifies the inbound port to which this URIMAP definition relates. If this definition is not given, the URIMAP definition relates to all TCPIPSERVICE definitions.

TEMPLATENAME(*value*)
Displays the name of a CICS document template that is used to form the static response. The template name can be up to 40 characters.

TRANSACTION(*value*)
Displays the 4-character name of an alias transaction to run the user application that composes a response.

USAGE(*value*)
Displays the usage for the URIMAP definition.

SERVER
The URIMAP definition is for CICS as an HTTP server. It maps the URL of an incoming HTTP request from a Web client to CICS application resources.

CLIENT
The URIMAP definition is for CICS as an HTTP client. It is used when CICS makes a client request for an HTTP resource on a server.

PIPELINE
The URIMAP definition is for a Web service. It specifies the processing that is to be performed on a request by which a client calls a Web service in CICS.

ATOM
The URIMAP definition is for an Atom feed. It is used for an incoming request for data that CICS makes available as an Atom feed.

USERID(*value*)
Displays the 8-character user ID under which the alias transaction is attached.

WEBSERVICE(*value*)
Displays the name of a Web service. This name can be the 1- to 8-character name of a WEBSERVICE resource definition or a name up to 32 characters representing a Web service generated by the CICS Web services assistant. It defines aspects of the runtime environment for a CICS application program deployed in a Web services setting.

CEMT INQUIRE VTAM

Inquire on the state of the connection between CICS and the z/OS Communications Server.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

The **INQUIRE VTAM** command inquires on the state of the connection between CICS and the Communications Server, the generic resource name, the type of Communications Server persistent sessions support, and the persistent session delay interval.

VTAM is now known as the z/OS Communications Server.

Input

Press the Clear key and type `CEMT INQUIRE VTAM` (or suitable abbreviations for the keywords). The resulting display is similar to the following example.

You can change various attributes in the following ways:

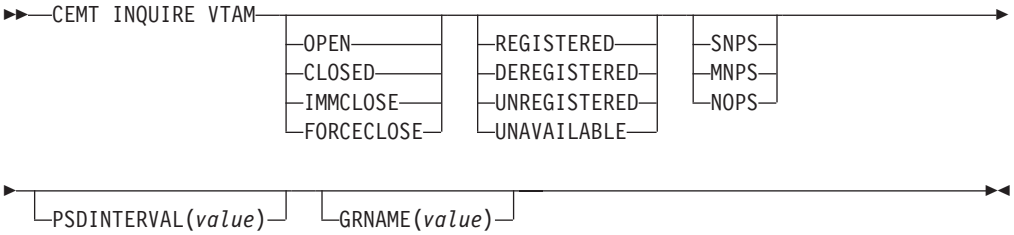
- Overtype your changes on the INQUIRE screen after tabbing to the appropriate field. See Overtyping a display.
- Use the CEMT SET VTAM command.

Sample screen

```
IN VT
RESULT - OVERTYPE TO MODIFY
  Vtam
  Openstatus( Open )
  Pstype(SNPS)
  Psdinterval( 000000 )
  Grstatus()
  Grname()
```

Figure 122. CEMT INQUIRE VTAM screen

CEMT INQUIRE VTAM



Displayed fields

GRNAME(*value*)
Displays the 8-character generic resource name under which this CICS region is registered to the Communications Server.

GRSTATUS(*value*)
Displays the status of generic resource registration. The values are as follows:

REGISTERED
CICS registered successfully as a member of the Communications Server generic resource named by the GRNAME option.

DEREGISTERED
CICS was registered as a member of the generic resource named by the GRNAME and has deregistered successfully.

UNREGISTERED
A generic resource name (GRNAME) was specified, but CICS has not made any attempt to register.

UNAVAILABLE
A generic resource name (GRNAME) was specified, but the version of the Communications Server to which CICS is connected does not support generic resources.

If CICS makes an unsuccessful attempt to register, the status is shown as UNREGISTERED, and the words REGISTER ERROR appear alongside.

If CICS makes as unsuccessful attempt to deregister, the status is shown as REGISTERED, and the words DEREGISTER ERROR appear alongside.

OPENSTATUS(*value***)**
Displays the status of the connection between CICS and the Communications Server. The values are as follows:

OPEN
A connection exists between CICS and the Communications Server.

CLOSED
The connection between CICS and the Communications Server has stopped.

You can reset this value by overtyping it with a different value or with one of the following:

IMMCLOSE
The connection between CICS and the Communications Server is in the process of closing following an EXEC CICS SET VTAM IMMCLOSE or CEMT SET VTAM IMMCLOSE command.

FORCECLOSE
The connection between CICS and the Communications Server is in the process of closing following an EXEC CICS SET VTAM FORCECLOSE or CEMT SET VTAM FORCECLOSE command.

PSDINTERVAL(*value***)**
Displays the persistent session delay interval, expressed in the form *hhmmss*. It specifies if and for how long Communications Server sessions are held in recovery-pending state if a failure occurs. You can reset this value by typing over it with a different value. The permitted range is 000000 to 235959 inclusive.

000000 If a failure occurs, sessions are ended. This value is the default, and means that persistent sessions support is not exploited. If **Pstype** is shown as NOPS, the persistent session delay interval must be zero.

HHMMSS
If a failure occurs, sessions are held in recovery-pending state by the Communications Server for up to the interval specified.

PSTYPE(*value***)**
Displays the type of Communications Server persistent sessions support for the CICS region. The values are as follows:

SNPS Single-node persistent sessions. Communications Server sessions can be recovered after a CICS failure and restart.

MNPS
Multinode persistent sessions. Communications Server sessions can also be recovered after a Communications Server or z/OS failure in a sysplex.

NOPS Communications Server persistent sessions support is not used for this CICS region.

VTAM
Indicates that this panel relates to a Communications Server inquiry.

CEMT INQUIRE WEB

Retrieve information about CICS web support.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

INQUIRE WEB returns information about the state of CICS web support.

Input

Press the Clear key and type `CEMT INQUIRE WEB` (or suitable abbreviations for the keywords). The resulting display is similar to the following example.

You can change various attributes in the following ways:

- Overtype your changes on the INQUIRE screen after tabbing to the appropriate field (see "Overtyping a display" on page 117).
- Use the CEMT SET WEB command.

Sample screen

```
IN WE
STATUS:  RESULTS - OVERTYPE TO MODIFY
 Web Gar( 00060 ) Tim( 00005 ) Loc
```

Figure 123. CEMT INQUIRE WEB screen

If you place the cursor against the result and press ENTER, CICS displays an expanded format, as shown in Figure 124.

```
I WEB
RESULT - OVERTYPE TO MODIFY
  Web
  Garbageint( 00060 )
  Timeoutint( 00005 )
```

Figure 124. The expanded display

CEMT INQUIRE WEB

Displayed fields

GARBAGEINT
The interval at which the web garbage collection task runs to clean up web 3270 state data for which the terminal timeout interval has expired. The interval is measured in minutes.

TIMEOUTINT
The period of time after which inactive web 3270 sessions are eligible for garbage collection. The period is measured in minutes.

CEMT INQUIRE WEBSERVICE

Retrieve information about an installed WEBSERVICE resource.

In the CICS Explorer, the Web services operations view provides a functional equivalent to this command.

Description

Use the **CEMT INQUIRE WEBSERVICE** command to retrieve information about an installed WEBSERVICE resource.

The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key to clear the screen. Start this transaction in one of two ways:

- Type `CEMT INQUIRE WEBSERVICE`. You get a screen that lists the current status.
- Type `CEMT INQUIRE WEBSERVICE` followed by a series of other attributes to limit the range of information that you require. So, for example, if you enter `CEMT I WEBSERVICE EN`, the resulting display shows you the details of only those resource types that are enabled.

To change various attributes:

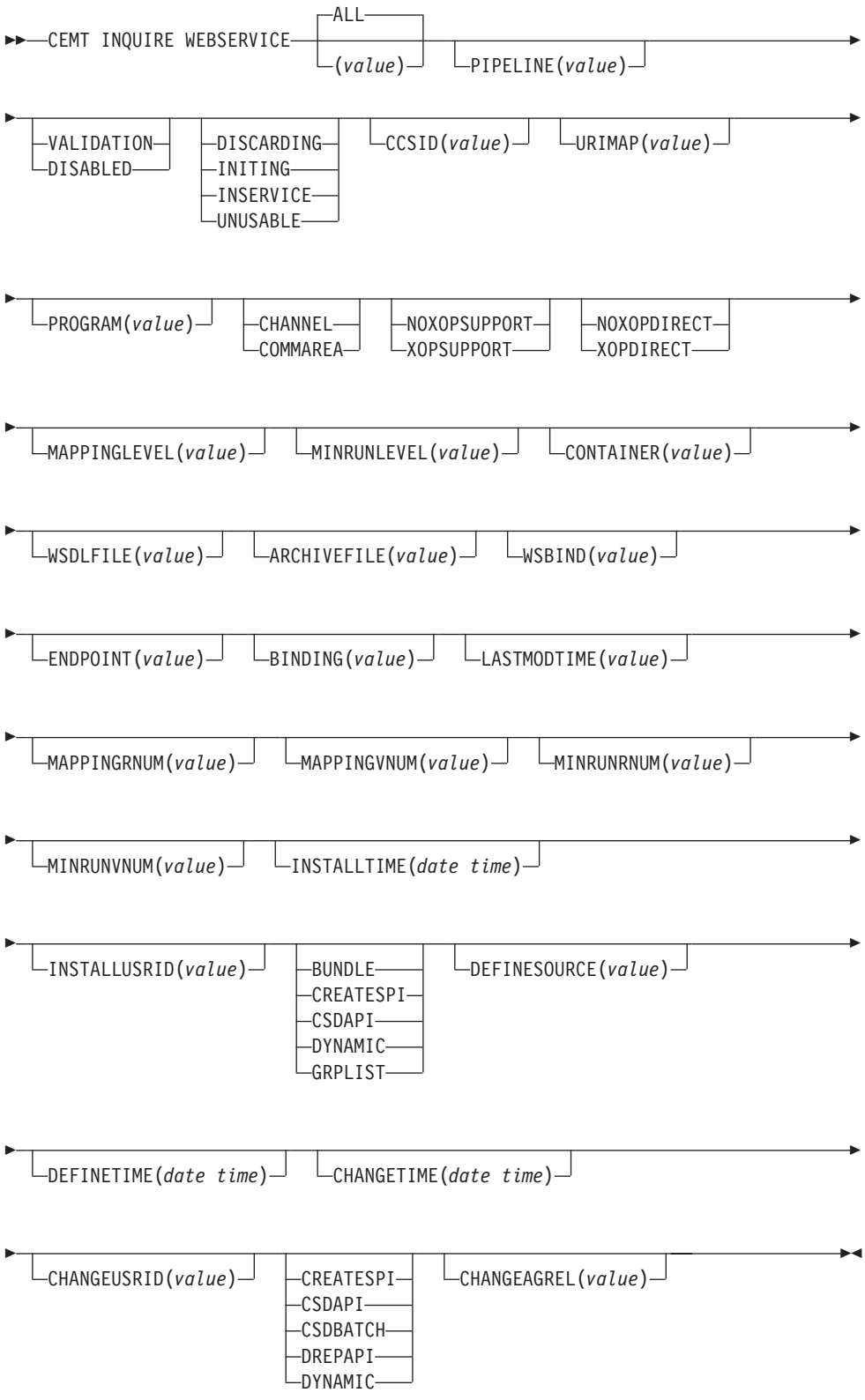
- Overtype your changes on the INQUIRE screen after tabbing to the appropriate field.
- Use the **CEMT SET WEBSERVICE** command.

ALL
Is the default. Information about all WEBSERVICE resources is displayed, unless you specify a selection of web services to be queried.

(value)
Is the 1 - 32 character name of an installed WEBSERVICE definition. The resource is up to 8 characters if installed using RDO and up to 32 characters if installed dynamically.

If you place the cursor against a specific entry in the list and press Enter, CICS displays an expanded format of all the attributes for the specified WEBSERVICE resource.

CEMT INQUIRE WEBSERVICE



Displayed fields

ARCHIVEFILE*(value)*
Displays the name and location (1-255 characters) of the archive that is associated with the web service. The archive contains one or more WSDL files. The name can be up to 255 characters long.

BINDING*(value)*
Displays the WSDL binding represented by the WEBSERVICE resource. This binding is one of (potentially) many that appear in the WEBSERVICE configuration file associated with the WEBSERVICE resource. The name can be up to 255 characters long.

CCSID(*value*)
Displays the CCSID that is used to encode data between the application program and the web service binding file at run time. This value is set using the optional **CCSID** parameter in the web services assistant when the web service binding file was generated. If the *value* is 0, the default CCSID for the CICS region that is specified by the **LOCALCCSID** system initialization parameter is used.

CHANGEAGENT(*value***)**
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

DYNAMIC
The resource definition was last changed by a PIPELINE scan.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

CONTAINER*(value)*
Displays the name of the container used if PGMINTERFACE returns a value of CHANNEL. The name can be up to 16 characters long.

DEFINESOURCE(*value***)**
Displays the source of the resource definition. The DEFINESOURCE value

depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

ENDPOINT*(value)*
Displays the endpoint URI of a remote WEBSERVICE resource. This endpoint URI is specified in the WSDL file for a remote web service. If a CICS application program is the service provider, the ENDPOINT is empty. The URI can be up to 255 characters long.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

BUNDLE
The resource was installed by a bundle deployment.

CREATESPI
The resource was installed by an **EXEC CICS CREATE** command.

CSDAPI
The resource was installed by a CEDA transaction or the programmable interface to DFHEDAP.

DYNAMIC
The resource was installed by using a PIPELINE scan.

GRPLIST
The resource was installed by **GRPLIST INSTALL**.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

LASTMODTIME(*value***)**
Displays the time when the deployed WSBind file on z/OS UNIX was last updated. The last-modified-time can be used to determine whether CICS has refreshed itself after an update is made to a WSBind file in the pickup directory.

For dynamically installed web services (those installed by the CICS scanning mechanism), the value of LASTMODTIME is the timestamp of the z/OS UNIX file pointed to by the WSBind definition, at the time the web service definition was last installed or updated. For statically-installed web services (those installed from a CSD or by CREATE WEBSERVICE), the value of LASTMODTIME is the timestamp of the WSBind z/OS UNIX file pointed to by the WEBSERVICE definition, at the time the WEBSERVICE resource was installed.

If you issue an **INQUIRE WEBSERVICE** command before a newly-installed or updated web service has fully initialized, the returned LASTMODTIME value is zero.

MAPPINGLEVEL(*value*)
Displays the mapping level that is used to convert data between language structures and web service description (WSDL) documents. The value of the mapping level is 1.0, 1.1, 1.2, 2.0, or 2.1. The default is to use a mapping level of 1.0.

MINRUNLEVEL(*value*)
Displays the minimum runtime level that is required to run the web service in CICS. The value of the runtime level is 1.0, 1.1, 1.2, 2.0, or 2.1.

PGMINTERFACE
Displays an indication of whether the CICS program that implements the web service expects input in a channel or in a commarea:

CHANNEL
The program expects input in a channel.

COMMAREA
The program expects input in a commarea.

PIPELINE*(value)*
Displays the name of the PIPELINE in which the web service is installed; that is, the name of the PIPELINE resource that contains this web service resource. The name can be up to 8 characters long.

PROGRAM*(value)*
Displays the name of a CICS program that implements the web service. If this WEBSERVICE resource represents a remote web service (that is, CICS is not the service provider), PROGRAM is empty. The name can be up to 8 characters long.

STATE*(value)*
Displays the state of the WEBSERVICE resource:

DISCARDING
A DISCARD command has been issued for the WEBSERVICE resource. The WEBSERVICE resource is quiescing before being discarded. It is not accepting new work, but is allowing currently executing work to complete.

INITING
The web service binding file, and the WSDL file, are being copied to the shelf.

INSERVICE
Resolution of the copy of the WSBIND file on the shelf has succeeded, and the WEBSERVICE is usable.

UNUSABLE
Copying of the WSBIND file on the shelf has failed, and the WEBSERVICE is unusable.

URIMAP*(value)*
Displays the name of a dynamically installed URIMAP definition, if one is associated with this WEBSERVICE resource. If the WEBSERVICE resource was not installed by performing the SCAN function on a PIPELINE resource, or if the WEBSERVICE resource represents a remote web service, the URIMAP definition is empty. The name can be up to 8 characters long.

VALIDATIONST*(value)*
Indicates whether full validation of SOAP messages is currently enabled for this WEBSERVICE resource:

VALIDATION
Full validation is enabled.

DISABLED
Full validation is disabled.

WSBIND*(value)*
Displays the name of the web service binding file. The name can be up to 255 characters long.

WSDLFILE*(value)*
Displays the name of the web service description file associated with the WEBSERVICE resource. The name can be up to 255 characters long.

XOPDIRECTST*(value)*
Indicates whether the web service can handle XOP documents and binary attachments in direct mode:

NOXOPDIRECT
The web service cannot handle XOP documents and binary attachments in direct mode, either because validation is switched on for the web service or because the web service implementation does not support the handling of XOP documents and binary attachments. Compatibility mode is used instead.

XOPDIRECT
The web service can handle XOP documents and binary attachments in direct mode.

XOPSUPPORTST*(value)*
Indicates whether the web service implementation can handle XOP documents and binary attachments.

NOXOPSUPPORT
The web service implementation cannot handle XOP documents and binary attachments.

XOPSUPPORT
The web service implementation can handle XOP documents and binary attachments. This case is true for any CICS-generated web service created by a level of CICS that supports MTOM/XOP.

CEMT INQUIRE WORKREQUEST

Retrieve information about work requests in the local CICS region.

Description

The **INQUIRE WORKREQUEST** command returns information about the various EJB and web service tasks that are started as a result of action by a request receiver. These tasks are also known as *work requests*.

You can use the **INQUIRE WORKREQUEST** command to inquire about a single work request in the local region:

- You can determine which transactions are associated with this work request
- You can correlate all transactions associated with this work request; for example, for accounting purposes

The **INQUIRE WORKREQUEST** command is based on the type of work being carried out. If you inquire on a specific work request and do not set a value for the work type, CICS uses IIOP as the default value. If you inquire on all work requests, the command returns work requests of all types.

Work requests are identified by tokens. Each token is unique in the local system for the lifetime of the request, but subsequently, it can be reused.

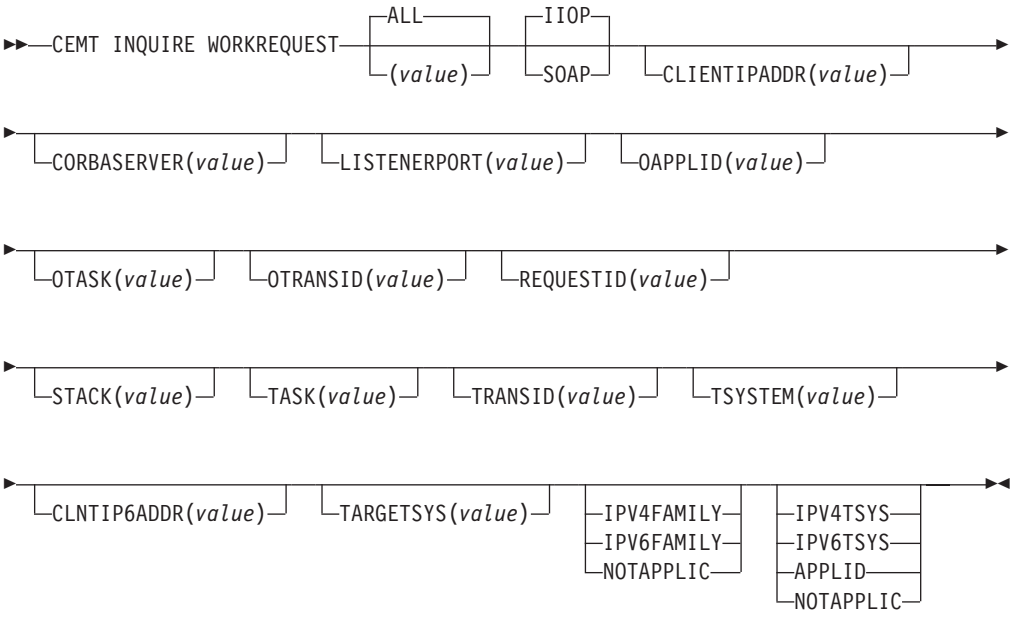
Input

Press the Clear key to clear the screen. There are two ways of starting this transaction:

- Type `CEMT INQUIRE WORKREQUEST`. You get a display that lists the current status.
- Type `CEMT INQUIRE WORKREQUEST` followed by as many of the other attributes as are necessary to limit the range of information that you require. So, for example, if you enter `cemt i wo co(MJW1)`, the resulting display shows you the details of only those work requests associated with the CORBASERVER definition named MJW1.

You cannot change the attributes displayed. Use the **CEMT SET WORKREQUEST** command to purge or force purge a particular work request.

CEMT INQUIRE WORKREQUEST



(*value***)**
An 8-character token to identify the work request. The valid characters for this token are the 16 hexadecimal digits, 0–9 and A–F. The token is generated by CICS. Normal usage begins with an **INQUIRE WORKREQUEST BROWSE** command, to identify the token which is of interest. Then the specific token can be used to issue INQUIRE and SET commands that relate to the task in question.

ALL
Is the default. As many work requests are displayed as are present in the system.

You cannot specify a list of identifiers, nor can you use the symbols * and + to specify a family of work requests.

When the character P or F (PURGE or FORCEPURGE) is entered in the action column CEMT initiates a **SET WORKREQUEST** command.

If you place the cursor against a specific entry in the list and press Enter, CICS displays an expanded format.

Displayed fields

CLNTIP6ADDR(*value*)
Displays the colon hexadecimal IPv6 address of the TCP/IP client that originated the request as a 39-character value. If the client address is in IPv4 format, the dotted decimal IPv4 address is returned.

CLIENTIPADDR(*value*)
Displays the dotted decimal IPv4 address of the client that originated the request as a 15-character value.

CLNTIPFAMILY(*value*)
Displays the address format of the IP address in CLIENTIPADDR and CLNTIP6ADDR. The CLNTIPFAMILY option is available using an expanded query only. Filtering interacts with the CLNTIPFAMILY option when you filter using wildcard characters. For example, if the value in CLNTIPFAMILY is IPV6FAMILY, you must use the colon character to retrieve an IP address. The possible values are as follows:

IPV4FAMILY
The address is an IPv4 address.

IPV6FAMILY
The address is an IPv6 address.

UNKNOWN
The IPRESOLVED option is not yet in use, or the address cannot be resolved.

CORBASERVER(*value*)
Displays the name of the CorbaServer as a 4-character value.

LISTENERPORT(*value*)
Displays the host port that received the request, as a 5-digit numeric value.

OAPPLID(*value*)
Displays the APPLID of the originating task as an 8-character value.

OTASK(*value*)
Displays the number of the task (RequestReceiver) that received the request, as a 7-digit numeric value.

OTRANSID(*value*)
Displays as a 4-character value, the transaction ID that received the request.

PURGETYPE (*input only field*)
Specifies whether a task is to be purged or force purged. The values are:
FORCEPURGE
The work request is terminated immediately. System integrity is not guaranteed. In some extreme cases, for example if a work request is force purged during backout processing, CICS terminates abnormally. If you want to terminate a work request but do not want to terminate CICS, use PURGE instead of FORCEPURGE.

PURGE
The work request is terminated, but termination occurs only when system and data integrity can be maintained.

REQUESTID(*value***)**
Displays the current request as a 7-digit numeric value. This value is sent by the client initiating the request and is used to associate the reply with the request. A client can be processing more than one request at any given moment.

STACK(*value***)**
Displays the stack level of this call as a 7-digit numeric value.

As each new request is received from the client the request receiver creates a stack and initializes it with a value of '1'. The stack is incremented by '1' every time a new bean is invoked, and decremented each time a bean which has incremented it terminates. So the value of the stack gives the current level within the EJB transaction at which this work request is executing. If a request is sent to a non-CICS EJB server then this field is NOT incremented.

TARGETSYS(*value***)**
Displays the APPLID of the target CICS system as a 50-byte field. This field can contain one of the following values:

- The colon hexadecimal TCP/IP address and port number of the target system, in the format `[::a:b:c:d]:port` where [] are X'BA' & X'BB' (code page 37).
- The dotted decimal TCP/IP address and port number of the target system. All IPv4 addresses are displayed as native IPv4 dotted decimal addresses, for example, `1.2.3.4`, irrespective of which type of address format is used. For information on accepted IPv4 formats, see the *CICS Internet Guide*.
- The z/OS Communications Server APPLID of the target system. The APPLID can be up to 8 characters followed by blanks.
- A string of blank characters. The target system is not CICS over MRO, the target system is not CICS over TCP/IP, or an error has occurred.
- A different value, which CICS captures but does not translate.

TASK(*value***)**
Displays the local task number, as a 7-character numeric value. It is translated to packed decimal, or in C to an unsigned char(4).

TRANSID(*value***)**
Displays the local transaction ID - 4 character

TSYSTEM(*value***)**
Displays the APPLID of the target CICS system as a 21-byte field. It can contain one of the following values:

- The dotted decimal IPv4 address and port number of the target system. All IPv4 addresses are displayed as native IPv4 dotted decimal addresses; for example, `1.2.3.4`, irrespective of which type of address format is used. For information on accepted IPv4 formats, see the *CICS Internet Guide*.
- Up to eight characters followed by blanks. In this case, these characters are the z/OS Communications Server APPLID of the target system.
- Another value. CICS does not know about any other possibilities. Any other value must be meaningful to other software at your installation which expects to work with the value obtained from this parameter of this CICS command.

TSYSTEM or TARGETSYS is needed when going outbound over TCP/IP. It is the only identification of the target system that is available.

TSYSTYPE(*value***)**
Displays the address format of the IP address in TSYSTEM and TARGETSYS. Here are the values:

IPV4 The TSYSTEM and TARGETSYS options contain a dotted decimal IPv4 address.

IPV6 The TARGETSYS option contains a colon hexadecimal IPv6 address. This value is returned only if an IPv6 address is returned in the TARGETSYS option.

APPLID
The TSYSTEM and TARGETSYS options contain an APPLID.

NOTAPPLIC
No entry is specified in the TSYSTEM or TARGETSYS option. Blanks are returned.

WORKREQUEST(*value***)**
Displays an 8-character token to identify the work request. The valid characters for this token are the 16 hexadecimal digits, 0–9 and A–F. The token is generated by CICS. Normal usage begins with an **INQUIRE WORKREQUEST BROWSE** command, to identify the token which is of interest. Then the specific token can be used to issue INQUIRE and SET commands that relate to the task in question.

WORKTYPE(*value***)**
Displays the type of work request that you are inquiring about.

IIOP Specifies that the work is being performed for an IIOP request.

SOAP Specifies that the work is being performed for a Web service request.

If you are inquiring on specific work requests and do not specify a value for WORKTYPE, IIOP is used as the default value.

CEMT INQUIRE XMLTRANSFORM

Retrieve information about an installed XMLTRANSFORM resource.

In the CICS Explorer, the XML transforms operations view provides a functional equivalent to this command.

The resource signature

You can use this command to display the resource signature fields. You can use these fields to manage resources by capturing details of when the resource was defined, installed, and last changed. For more information, see Auditing resources in the Resource Definition Guide. The resource signature fields are CHANGEAGENT, CHANGEAGREL, CHANGETIME, CHANGEUSRID, DEFINESOURCE, DEFINETIME, INSTALLAGENT, INSTALLTIME, and INSTALLUSRID. See Summary of the resource signature field values in the Resource Definition Guide for detailed information about the content of the resource signature fields.

Input

Press the Clear key to clear the screen. There are two ways of starting this transaction:

- Type `CEMT INQUIRE XMLTRANSFORM`. You get a screen that lists the current status.
- Type `CEMT INQUIRE XMLTRANSFORM` followed by as many of the other attributes as are necessary to limit the range of information that you require. So, for example, if you enter `CEMT INQUIRE XMLTRANSFORM EN`, the resulting display shows you the details of only those resources that are enabled.

To change various attributes, you can:

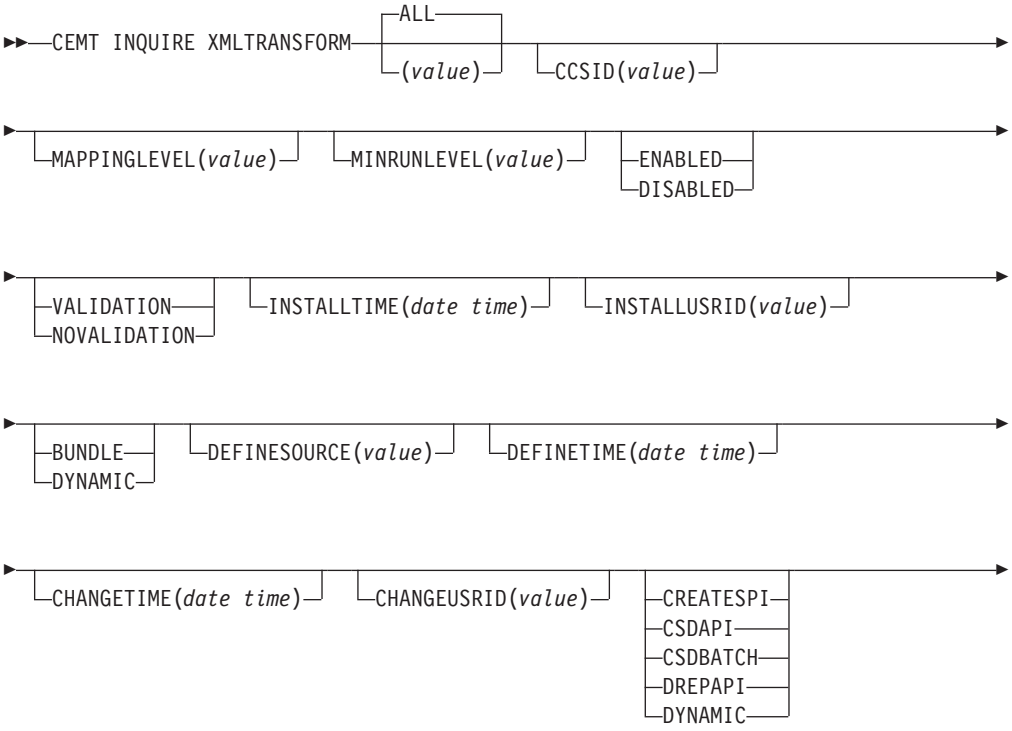
- Overtype your changes on the INQUIRE screen after tabbing to the appropriate field.
- Use the **CEMT SET XMLTRANSFORM** command.

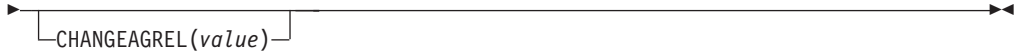
ALL
is the default. Information about all XMLTRANSFORM resources is displayed, unless you specify a selection of resources to be queried.

(value)
is the name (1-32 characters) of an installed XMLTRANSFORM resource.

If you place the cursor against a specific entry in the list and press ENTER, CICS displays an expanded format of all the attributes for the specified XMLTRANSFORM resource.

CEMT INQUIRE XMLTRANSFORM





Displayed fields

CCSID(*value*)
Displays the CCSID that is used to encode data the character data in the application data structure at run time. This value is set using the optional **CCSID** parameter in the XML assistant when the XML binding file is generated. If the *value* is 0, CICS uses the default CCSID that is specified by the **LOCALCCSID** system initialization parameter.

CHANGEAGENT(*value***)**
Displays a value that identifies the agent that made the last change to the resource definition. You cannot use CEMT to filter on some of these values because they are duplicated. The possible values are as follows:

CREATESPI
The resource definition was last changed by an **EXEC CICS CREATE** command.

CSDAPI
The resource definition was last changed by a CEDA transaction or the programmable interface to DFHEDAP.

CSDBATCH
The resource definition was last changed by a DFHCSDUP job.

DREPAPI
The resource definition was last changed by a CICSPlex SM BAS API command.

DYNAMIC
The resource was defined by an ATOMSERVICE resource.

CHANGEAGREL(*value***)**
Displays the 4-digit number of the CICS release that was running when the resource definition was last changed.

CHANGETIME(*date time***)**
Displays the date and time when the resource definition was last changed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

CHANGEUSRID(*value***)**
Displays the 8-character user ID that ran the change agent.

DEFINESOURCE(*value***)**
Displays the source of the resource definition. The DEFINESOURCE value depends on the CHANGEAGENT option. For details, see Summary of the resource signature field values in the Resource Definition Guide.

DEFINETIME(*date time***)**
Displays the date and time when the resource was created. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

ENABLESTATUS*(value)*
Displays the status of the installed XMLTRANSFORM resource. The possible values are:

ENABLED
The XMLTRANSFORM resource is enabled for use.

DISABLED
The XMLTRANSFORM resource is disabled.

INSTALLAGENT(*value***)**
Displays a value that identifies the agent that installed the resource. The possible values are as follows:

BUNDLE
The resource was installed by a bundle deployment.

DYNAMIC
The resource was installed by an ATOMSERVICE resource.

INSTALLTIME(*date time***)**
Displays the date and time when the resource was installed. The format of the date depends on the value that you selected for the DATFORM system initialization parameter for your CICS region. The format of the time is hh:mm:ss.

INSTALLUSRID(*value***)**
Displays the 8-character user ID that installed the resource.

MAPPINGLEVEL(*value*)
Displays the mapping level that was used when the XML binding file was generated. The value of the mapping level is 1.0, 1.1, 1.2, 2.0, 2.1, 2.2, or 3.0.

MINRUNLEVEL(*value*)
Displays the minimum runtime level that is required to use the XMLTRANSFORM resource in CICS. The value of the runtime level is 3.0.

VALIDATIONST*(value)*
Displays whether full validation is currently enabled for this XMLTRANSFORM resource:

VALIDATION
Full validation is enabled

NOVALIDATION
Full validation is not enabled.

XMLSCHEMA*(value)*
Displays the name of the XML binding file. The name can be up to 255 characters long.

XSDBIND*(value)*
Displays the name of the XML binding file. The name can be up to 255 characters long.

CEMT PERFORM commands

Use the **CEMT PERFORM** commands to perform actions in CICS, including scanning directories in z/OS UNIX, taking system dumps, gathering statistics, and shutting down CICS.

Descriptions of the **CEMT PERFORM** commands follow in alphabetic order. The operands for each of the commands are also presented in alphabetic order.

CEMT PERFORM CLASSCACHE

Initialize and terminate the shared class cache that is used by pooled JVMs in the CICS region.

In the CICS Explorer, the JVM classcache operations view provides a functional equivalent to this command.

Description

You can use the **CEMT PERFORM CLASSCACHE** command to start, reload, and terminate the shared class cache. This class cache is used by pooled JVMs in the CICS region. You cannot use this command for class caches that are used by JVM servers.

While you are performing one of these operations, you can use other options on the command to set attributes of the shared class cache, as follows:

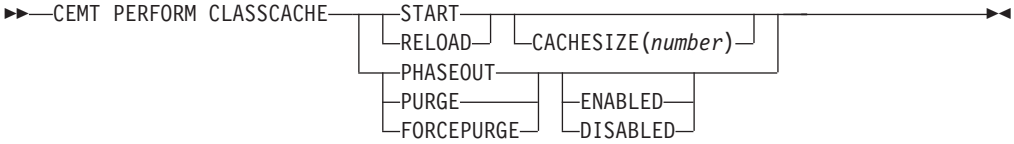
- When you are initializing the shared class cache (using the START or RELOAD option), you can change its size (CACHESIZE option). The new setting overrides the size originally set by the **JVMCCSIZE** system initialization parameter .
- When you are terminating the shared class cache (using the PHASEOUT, PURGE, or FORCEPURGE option), you can set the status of autostart (AUTOSTARTST option). If you do not want the shared class cache to start up again until you enter an explicit command, you can use this option to ensure that autostart is disabled. The new setting overrides the autostart status that was originally set by the **JVMCCSTART** system initialization parameter.

When you change the size or autostart status of the shared class cache while CICS is running, subsequent warm or emergency CICS restarts use the changed setting. However, the shared class cache normally persists across warm or emergency restarts, so the changed setting normally has no effect in this situation. If the system is initial or cold started, the settings are always taken from the relevant system initialization parameters.

When you terminate the shared class cache and autostart is enabled, a shared class cache is started as soon as a new pooled JVM requests it. If autostart is disabled, you must manually start the shared class cache. If you do not restart the shared class cache, requests to run a Java application in a pooled JVM whose profile requires the use of the shared class cache fail. Specify the AUTOSTARTST option to set the status of autostart.

Syntax

CEMT PERFORM CLASSCACHE



Options

FORCEPURGE

Stops and deletes the shared class cache. All tasks using pooled JVMs that are using the shared class cache are terminated by the **SET TASK FORCEPURGE**

mechanism, and the JVMs are terminated. The shared class cache is deleted when all the pooled JVMs that were using it have been terminated.

PHASEOUT
Stops and deletes the shared class cache. All pooled JVMs using the shared class cache are marked for deletion. The JVMs are deleted when they finish running their current Java programs. The shared class cache is deleted when all the JVMs that were using it have been terminated.

When you have issued one command to terminate the shared class cache, you can repeat the **CEMT PERFORM CLASSCACHE** command to make a stronger attempt to purge the tasks using the pooled JVMs (for example, by using PURGE instead of PHASEOUT). Only repeat the command if autostart for the shared class cache is disabled. If autostart is enabled and you repeat the command to terminate the shared class cache, the command might operate on the new shared class cache that has been started by the autostart facility, and terminate it.

PURGE
Stops and deletes the shared class cache. All tasks using pooled JVMs that are using the shared class cache are terminated by the SET TASK PURGE mechanism, and the JVMs are terminated. The shared class cache is deleted when all the JVMs that were using it have been terminated.

RELOAD
Reloads the shared class cache. Perform this operation to create a class cache when the status of the shared class cache is STARTED (the option only works if the shared class cache has been started).

If specified, the CACHESIZE is used in the creation of the cache. JVMs, both those that are already allocated to tasks and those that are allocated to tasks after you issue the command, continue to use the existing shared class cache until the new shared class cache is ready. When the new shared class cache is ready, subsequent requests for JVMs are given a JVM that uses the new cache. These new JVMs are started when they are requested by applications, and they replace the JVMs that are using the old shared class cache. The JVMs that are using the old shared class cache are allowed to finish running their current Java programs, and then they are terminated. The old shared class cache is deleted when all the JVMs that are using it have been terminated.

START
Starts the shared class cache. Perform this operation to create a new class cache when the status of the shared class cache is STOPPED. If specified, the CACHESIZE is used in the creation of the cache.

The additional options that you can use on the **PERFORM CLASSCACHE** command to set attributes of the shared class cache are:

CACHESIZE(*number***)**
You can use CACHESIZE when you are using the START or RELOAD options. CACHESIZE specifies the size, in bytes, of the shared class cache that is to be started, or the new shared class cache that is to be reloaded. You can specify the size in bytes, or in kilobytes (K), megabytes (M) or gigabytes (G). For example, specifying `20M` sets the size of the shared class cache to 20 MB. The number you specify can be up to 20 digits. The minimum size that you can specify is `8M` (8388608 bytes).

DISABLED
If you are using the PHASEOUT, PURGE, or FORCEPURGE options, you can set the status of autostart for the shared class cache to DISABLED.

When autostart is disabled, if the shared class cache has been stopped or has not yet been started on this CICS run, an explicit **PERFORM CLASSCACHE** command is required to start it. If the status of the shared class cache is STOPPED and autostart is disabled, and CICS receives a request to run a Java application in a JVM with a profile that requires the use of the shared class cache, the application cannot run.

ENABLED
If you are using the PHASEOUT, PURGE, or FORCEPURGE options, you can set the status of autostart for the shared class cache to ENABLED.

When autostart is enabled, if the shared class cache has been stopped or has not yet been started on this CICS run, the shared class cache is started as soon as CICS receives a request to run a Java application in a JVM with a profile that requires the use of the shared class cache.

CEMT PERFORM CORBASERVER

Publish or retract all beans that are installed in the specified CorbaServer and scan z/OS UNIX for new or updated deployed JAR files.

Description

You can use the **PERFORM CORBASERVER** command to publish or retract all beans installed in the specified CorbaServer and the Generic Factory Interoperable Object Reference (GenFacIOR) of the specified Corbaserver. You can also scan the deployed JAR file directory of the CorbaServer for new or updated deployed JAR files.

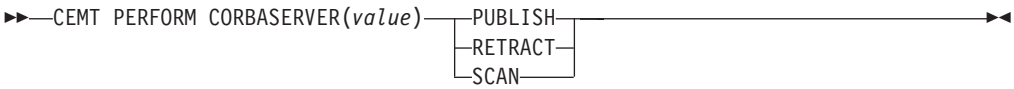
If no action is specified, the command has no effect. You can issue a **PERFORM CORBASERVER** command only when the CorbaServer is in a steady state; that is, when it is in ENABLED or DISABLED state, but not when it is in ENABLING, DISABLING, or DISCARDING state.

In addition to the traditional DFSMS-managed data sets, such as partitioned data sets (PDS), CICS requires access to z/OS UNIX System Services and its files to launch a JVM. The CICS region user ID must be authorized to z/OS UNIX System Services to enable CICS to access the required directories listed in the JVM profile and JVM properties file.

A failure to perform the requested action is reported by returning a condition and RESP2 value from the command and, in certain cases, by writing messages to CSMT. The requested action might have been partially successful since a failure does not cause the effects of the command to be backed out. Also, failure does not necessarily stop the command from continuing in order to increase the extent to which it was partially successful.

Syntax

CEMT PERFORM CORBASERVER



Options

CORBASERVER (*value*)
Specifies the 4-character CorbaServer name.

PUBLISH

Publishes all beans installed in the specified CorbaServer and the Generic Factory Interoperable Object Reference (Generic factory) of the specified Corbaserver.

Publishing a bean means binding a reference to the home of the bean in the namespace. The naming context in which the bean is bound is named, relative to the initial context defined for the CICS region, using a concatenation of the JNDI PREFIX attribute of the CorbaServer and the name of the bean.

The Generic factory is bound with the name GenericFactory concatenated to the PREFIX attribute of the Corbaserver.

RETRACT
Indicates that all the beans installed in the CorbaServer are to be retracted. Retracting a bean means unbinding a reference to the home of the bean from the namespace. The naming context in which the bean is bound is named, relative to the initial context defined for the CICS region, using a concatenation of the JNDI prefix attribute of the CorbaServer and the name of the bean.

SCAN
scans the CorbaServer's deployed JAR file directory (also known as the *pickup directory*) for new or updated deployed JAR files.

If CICS finds any new deployed JAR files in the pickup directory, it copies them to its shelf directory and dynamically creates and installs DJAR definitions for them.

If CICS finds any updated deployed JAR files in the pickup directory, it updates both the LASTMODTIME, DATESTAMP, and TIMESTAMP attributes of the installed DJAR definitions and the shelf copies of the deployed JAR files.

Note: CICS ignores any deployed JAR files in the pickup directory that have the same name and the same date and time stamps as currently-installed DJAR resources. A deployed JAR file with the same name but a later date-and-time stamp than an installed DJAR is treated as an update.

If you receive a "DJAR CONFLICT" message, it means that one or more deployed JAR files have failed to install due to a naming conflict. This error can occur for two reasons:

1. The name of a DJAR definition created by the scanning mechanism conflicts with that of a previously-installed dynamic definition that targets a different CorbaServer. The scanned-in definition cannot be treated as an update, because the installed definition refers to a different CorbaServer.
2. The name of a DJAR definition created by the scanning mechanism conflicts with that of a statically-installed definition. You cannot use the scanning mechanism to update a static DJAR definition.

To avoid this error:

1. If your CICS region contains multiple CorbaServers, use a different pickup directory for each CorbaServer.
2. Do not place identically-named deployed JAR files into multiple pickup directories in the same CICS region.

3. If you use static DJAR definitions, do not give them names that might conflict with those created by the scanning mechanism.

CEMT PERFORM DELETSHIPPED

Cause the CICS timeout delete mechanism to be invoked immediately.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

The CICS timeout delete mechanism removes any shipped terminal definitions that have not been used for longer than the time specified on the DSHIPIDL system initialization parameter, or on a subsequent SET DELETSHIPPED IDLE command. The interval between scheduled invocations of the mechanism is specified on the DSHIPINT system initialization parameter, or on a subsequent SET DELETSHIPPED INTERVAL command.

PERFORM DELETSHIPPED invokes timeout delete immediately.

Note:

1. Shipped definitions are not deleted if there is an automatic initiate descriptor (AID) associated with the terminal.
2. Issuing a PERFORM DELETSHIPPED command does **not** reset the start of the time interval until the next invocation of the timeout delete mechanism. In other words, causing the timeout delete mechanism to be invoked immediately does not affect the time remaining until the next scheduled invocation.

Syntax

CEMT PERFORM DELETSHIPPED

CEMT PERFORM DJAR

The PERFORM DJAR command performs certain actions against a specified deployed JAR file. If no action is specified, the command has no effect.

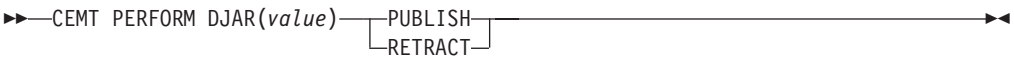
Description

In addition to the traditional DFSMS-managed data sets, such as partitioned data sets (PDS), CICS requires access to z/OS UNIX System Services and its files in order to launch a JVM. This requires that the CICS region userid is authorized to z/OS UNIX System Services to enable CICS to access the required z/OS UNIX directories listed in the JVM profile and JVM properties file.

A failure to perform the requested action will be reported by returning a condition and RESP2 value from the command and, in certain cases, by writing messages to CSMT. The requested action may have been partially successful since a failure does not cause the effects of the command to be backed out. Also, failure does not necessarily stop the command from continuing in order to increase the extent to which it was partially successful.

Syntax

CEMT PERFORM DJAR



Options

DJAR (value)
specifies the 1–32 character name of a deployed JAR file.

PUBLISH
indicates that all beans installed in the named deployed JAR file are to be published. Publishing a bean means binding a reference to the home of the bean in the namespace. The naming context in which the bean is bound is named, relative to the initial context defined for the CICS region, using a concatenation of the JNDI prefix attribute of the DJAR's CorbaServer and the name of the bean.

RETRACT
indicates that all the beans installed in the deployed JAR file are to be retracted. Retracting a bean means unbinding a reference to the home of the bean from the namespace. The naming context in which the bean is bound is named, relative to the initial context defined for the CICS region, using a concatenation of the JNDI prefix attribute of the DJAR's CorbaServer and the name of the bean.

CEMT PERFORM DUMP|SNAP

Obtain a CICS system dump.

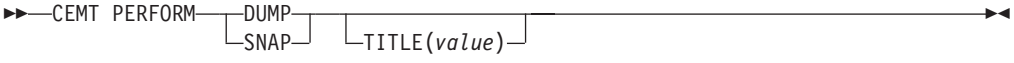
Description

To get a CICS system to dump and continue processing, you can use either the **CEMT PERFORM DUMP** command or the **CEMT PERFORM SNAP** command. Both commands produce a "snapshot" of the CICS system.

The use of **CEMT PERFORM DUMP** or **CEMT PERFORM SNAP** prevents all other tasks from running. Consequently, application or service users might experience delays in response.

Syntax

CEMT PERFORM DUMP



Options

DUMP
Specifies that a system dump is to be produced.

SNAP
Specifies that CICS is to issue the **MVS SDUMP** macro to produce an SDUMP (system dump) of the CICS region.

TITLE (*value*)
Specifies a title of up to 32 characters; it is added at the beginning of the dump. If your title includes spaces, you must enclose the whole title within single quotation marks (' ').

CEMT PERFORM ENDAFFINITY

End an affinity owned by CICS.

In the CICS Explorer, the ISC/MRO connections operations view provides a functional equivalent to this command.

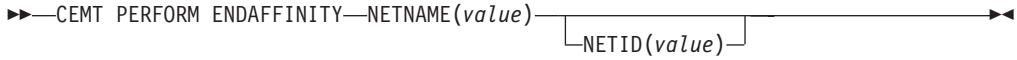
Description

Where CICS is a member of a z/OS Communications Server generic resource group, PERFORM ENDAFFINITY instructs z/OS Communications Server to end an affinity owned by CICS, whether or not the connection has been deleted. If the connection has not been deleted, it must be out of service and have no recovery information outstanding—that is, its RECOVSTATUS (as displayed by CEMT INQUIRE/SET CONNECTION) must be NORECOVDATA.

Generic resources and affinities are described in the *CICS Intercommunication Guide*.

Syntax

CEMT PERFORM ENDAFFINITY



Options

NETID (*value*)
specifies the 8-character name by which the network containing the connected LU is known to z/OS Communications Server.

If you do not specify a NETID, CICS takes the value from the installed connection, if it exists. If you do not specify a NETID and the connection does not exist, the command fails.

NETNAME (*value*)
specifies the 8-character APPLID of the connected LU. If the connected LU is a member of a generic resource, you must specify its member name, not the generic resource name.

Note: There is no facility in z/OS Communications Server for inquiring on affinities, so CICS has no certain knowledge that an affinity exists for a given connection. Whenever there is a possibility that an affinity has been created that you must end explicitly, CICS issues message DFHZC0177. This message gives the NETNAME and NETID to be passed to z/OS Communications Server.

If a request to end an affinity is rejected by z/OS Communications Server because no such affinity exists, CICS issues message DFHZC0181. This may mean either that you entered the NETNAME or NETID incorrectly, or that you (or CICS) was wrong in supposing that an affinity existed.

CEMT PERFORM JVMPOOL

Start and terminate JVMs in the JVM pool.

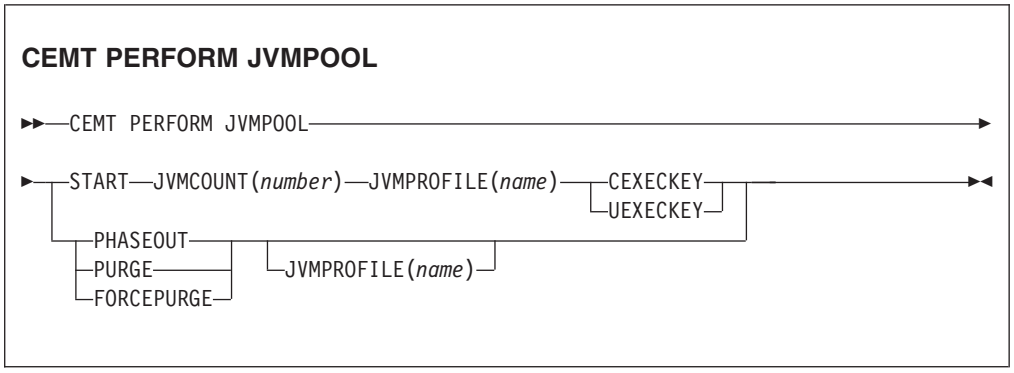
In the CICS Explorer, the JVM pools operations view provides a functional equivalent to this command.

Description

You can use the CEMT PERFORM JVMPOOL command to start JVMs with your chosen JVM profile and execution key. You can also use the command to terminate all or some of the JVMs in the pool, in order to implement changes to JVM profiles, or to add new application classes.

The CEMT PERFORM JVMPOOL TERMINATE command does not terminate the shared class cache. If the CICS region has a shared class cache, and you want to terminate it, use the CEMT PERFORM CLASSCACHE command to do this.

Syntax



Options

CEXECKEY
Specifies that the JVMs to be started are to run in CICS key.

FORCEPURGE
All tasks using JVMs with the profile specified by the Jvmprofile option, or (if Jvmprofile is not specified) using all JVMs in the pool, are terminated by the SET TASK FORCEPURGE mechanism, and the JVMs are terminated.

JVMCOUNT(*number*)
Specifies the number of JVMs to be started. If the number of JVMs you request, added to the number of JVMs that already exist, would mean exceeding the MAXJVMTCBS limit for the CICS region, CICS does not start any JVMs.

JVMPROFILE(*name*)
Specifies the 8-character name of a JVM profile. The name is case-sensitive, and you must enter it using the same combination of upper and lower case characters that is present in the z/OS UNIX file name. If you need to enter the name of a JVM profile in mixed case when you are using the CEMT transaction, ensure that the terminal you use is correctly configured, with upper case translation suppressed.

Jvmprofile is required with Start, to specify the profile for the JVMs that are to be started.

Jvmprofile can optionally be specified with Phaseout, Purge, or Forcepurge, to mean that only JVMs with this profile should be terminated. Without the Jvmprofile option, these options act on all the JVMs in the pool.

PHASEOUT
JVMs with the profile specified by the Jvmprofile option, or (if Jvmprofile is not specified) all JVMs in the pool, are marked for deletion. The JVMs are deleted when they finish running their current Java program.

PURGE
All tasks using JVMs with the profile specified by the Jvmprofile option, or (if Jvmprofile is not specified) using all JVMs in the pool, are terminated by the SET TASK PURGE mechanism, and the JVMs are terminated.

START(*number***)**
Specifies a number of JVMs to be started. Jvmcount, Jvmprofile and either Cexeckey or Uexeckey are required with Start, to specify the number, profile and execution key for the JVMs. CICS starts the JVMs asynchronously.

You cannot specify Start when the JVM pool's status is set to Disabled.

UEXECKEY
Specifies that the JVMs to be started are to run in user key. If the system initialization parameter STGPROT=NO is in effect, this is ignored, and the JVMs are started in CICS key.

CEMT PERFORM PIPELINE

Use the PERFORM PIPELINE command to initiate a scan of the Web service binding directory that is specified in the WSBIND attribute of the PIPELINE definition.

In the CICS Explorer, the Pipelines operations view provides a functional equivalent to this command.

Description

If the WSBIND attribute is not specified, there is nothing to scan, and control returns to your program. If the directory location specified is valid, CICS examines the Web service binding files in the directory to determine if they should be installed into the system:

- CICS installs any files it finds that have not been installed already.
- If a file has been installed already, but the file in the directory is newer than the one currently in use, the one that is in use is discarded, and the newer file is installed in its place.

If, for any reason, CICS fails to install an individual Web service binding file, processing continues with the remaining files in the directory. When the scan completes, the PIPELINE is available for use with whichever of the binding files were installed successfully.

A failure to perform the requested action will be reported by returning a condition and RESP2 value from the command and, in certain cases, by writing messages to CSMT. The requested action may have been partially successful since a failure does not cause the effects of the command to be backed out. Also, failure does not necessarily stop the command from continuing in order to increase the extent to which it was partially successful.

Syntax

CEMT PERFORM PIPELINE

Options

PIPELINE (*value***)**
specifies the 8-character PIPELINE name.

SCAN
Scan the PIPELINE's Web service binding directory

CEMT PERFORM RESET

Synchronize the CICS date and time-of-day with the system date and time-of-day.

Description

To ensure that the correct local time is used by all CICS functions, **use the CEMT PERFORM RESET command immediately whenever you alter the system date or the time-of-day in the MVS TOD clock while a CICS region is running.** For example, when setting clocks forward or back an hour to adjust for Summer and Winter time, use this command to ensure that CICS immediately updates the correct local time.

Note: To ensure that your applications are receiving the correct time, always use **CEMT PERFORM RESET** whenever you alter the MVS TOD clock. CICS only obtains and stores the *local time offset*, at start up, and when the CEMT PERFORM RESET command executes.

Syntax

CEMT PERFORM RESET

CEMT PERFORM SECURITY

Rebuild the in-store RACF resource profiles. The RACF resource profiles are rebuilt by reissuing the RACLIST macro for the active class names.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Syntax

CEMT PERFORM SECURITY

If the external security manager is RACF 2.1 or an equivalent product, the PERFORM SECURITY command is not required to refresh the classes used for

resource checks. A message of NOT REQUIRED is displayed and no action is taken.

Options

REBUILD
is an optional keyword. It does not alter the action of the transaction.

CEMT PERFORM SHUTDOWN

Shut down the CICS region.

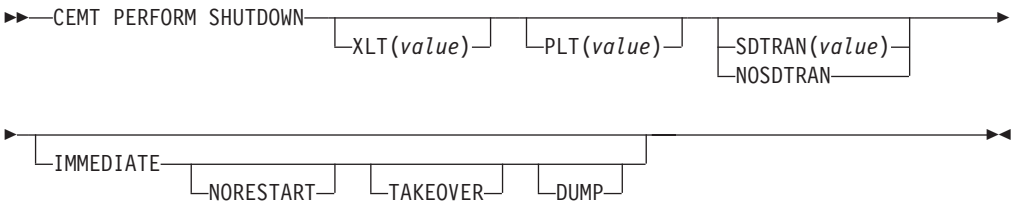
In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

The shutdown can be either controlled or immediate. If a shutdown (either normal or immediate) hangs, a subsequent SHUTDOWN IMMEDIATE resumes the shutdown process.

Syntax

CEMT PERFORM SHUTDOWN



Options

DUMP
A dynamic storage dump is produced upon completion of the termination process.

IMMEDIATE
Specifies that the system is shut down immediately, terminating all active tasks and SNA sessions. If IMMEDIATE is not specified, all tasks are allowed to finish, and SNA sessions are allowed to terminate normally.

NORESTART
Specifies that, if the CICS region is registered to MVS Automatic Restart Manager (ARM), it is to be deregistered as part of shutdown and not restarted automatically.

NOSDTRAN
No shutdown assist transaction is to be run at CICS shutdown.

The default shutdown transaction, if neither SDTRAN nor NOSDTRAN are specified, is CESD.

PLT (*value*)
Specifies the 2-character suffix of a program list table (PLT) to be used for the shutdown.

SDTRAN
The 4-character identifier of the shutdown assist transaction

TAKEOVER
Specifies that this CICS region should be closed down, and the alternate CICS region is to take over. This is valid only if the XRF=YES system initialization parameter was specified for CICS startup.

XLT (*value***)**
Specifies the 2-character suffix of a transaction list table (XLT) to be used for the shutdown.

Transactions listed in the specified XLT can be initiated after the SHUTDOWN request, and before the system quiesces. The default XLT should contain an entry for the master terminal transaction. This ensures that you can issue an immediate shutdown if, for example, a task is found to be suspended after an ordinary (nonimmediate) shutdown has been requested.

Table 5 shows the effect of the TAKEOVER and IMMEDIATE options on the active CICS system.

Table 5. Effect of the TAKEOVER and IMMEDIATE options on the active CICS system

	CEMT PERFORM SHUTDOWN		
	(normal)	***TAKEOVER***	***IMMEDIATE***
Tasks	finish	finish	terminate
SNA session (XRF)	unbind	leave	leave
SNA session (non-XRF)	unbind	unbind	unbind

Table 6 shows the effect of the TAKEOVER and IMMEDIATE options on the alternate CICS system.

Table 6. Effect of the TAKEOVER and IMMEDIATE options on the alternate CICS system

	CEMT PERFORM SHUTDOWN		
	(normal)	***TAKEOVER***	***IMMEDIATE***
CICS alternate	shuts down	takes over	takes over

CEMT PERFORM STATISTICS

Write the statistics for all, or a named resource type immediately to the SMF data set, rather than wait for the current statistics-gathering interval to expire.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

The recording is made irrespective of the ON or OFF setting of statistics. See also “CEMT INQUIRE STATISTICS” on page 302 and “CEMT SET STATISTICS” on page 498 for a description of the CEMT INQUIRE and SET STATISTICS commands.

You can request either statistics for all resources (ALL), or statistics for individual resources in various combinations by selecting one or more keywords from the list.

The message 'STATS MISSING' appears when the resource type is either not functioning or not available. As much data is recorded as possible.

See the *CICS Operations and Utilities Guide* for information about printing statistics using the CICS-supplied utility program DFHSTUP and for information about the reports produced by DFHSTUP.

Syntax

CEMT PERFORM STATISTICS

Options

ALL
Statistics for all resources are to be written immediately to the SMF data set.

ALL RESETNOW resets the counters for the collection of statistics. RESETNOW only works if the ALL operand is set, that is, ALL RESETNOW.

ATOMSERVICE
Atom feed statistics are to be written immediately to the SMF data set.

AUTOINSTALL
Autoinstall statistics are to be written immediately to the SMF data set.

BEAN
Enterprise bean statistics are to be written immediately to the SMF data set.

BUNDLE
Bundle statistics are to be written immediately to the SMF data set.

CAPTURESPEC
Capture specification statistics are to be written immediately to the SMF data set.

CONNECTION
Connection statistics are to be written immediately to the SMF data set.

CORBASERVER
CorbaServer statistics are to be written immediately to the SMF data set.

DB2
CICS DB2 statistics are to be written immediately to the SMF data set.

DISPATCHER
Dispatcher statistics are to be written immediately to the SMF data set.

DOCTEMPLATE
Document template statistics are to be written immediately to the SMF data set.

ENQUEUE
Enqueue statistics are to be written immediately to the SMF data set.

EPADAPTER
EP adapter statistics are to be written immediately to the SMF data set.

EVENTBINDING
Event binding statistics are to be written immediately to the SMF data set.

EVENTPROCESS
Event processing statistics are to be written immediately to the SMF data set.

FEPI
FEPI statistics are to be written immediately to the SMF data set.

FILE
File control statistics are to be written immediately to the SMF data set.

IPCONN
IPCONN statistics are to be written immediately to the SMF data set.

JOURNALNAME
CICS journal statistics are to be written immediately to the SMF data set.

Note: The JOURNALNUM resource name is obsolete, and replaced by the JOURNALNAME resource name.

JVMPOOL
JVM pool statistics are to be written immediately to the SMF data set.

JVMPROFILE
JVM profile statistics are to be written immediately to the SMF data set.

JVMPROGRAM
JVM program statistics are to be written immediately to the SMF data set.

JVMSERVER
JVMSERVER statistics are to be written immediately to the SMF data set.

LIBRARY
LIBRARY statistics are to be written immediately to the SMF data set.

LSRPOOL
LSRPOOL statistics are to be written immediately to the SMF data set.

MONITOR
Monitoring statistics are to be written immediately to the SMF data set.

MQCONN
WebSphere MQ Connection statistics are to be written immediately to the SMF data set.

PIPELINE
PIPELINE statistics are to be written immediately to the SMF data set.

PROGAUTO
Autoinstalled program statistics are to be written immediately to the SMF data set.

PROGRAM
Program statistics are to be written immediately to the SMF data set.

PROGRAMDEF
Program definition statistics are to be written immediately to the SMF data set.

RECORD
RECORD is an optional keyword and does not alter the action of the transaction.

RECOVERY
Recovery statistics are to be written immediately to the SMF data set.

REQUESTMODEL
Requestmodel statistics are to be written immediately to the SMF data set.

RESETNOW
The statistics counters are to be reset.

STATS
Statistics domain statistics are to be written immediately to the SMF data set.

STORAGE
Storage manager statistics are to be written immediately to the SMF data set.

STREAMNAME
Log stream statistics are to be written immediately to the SMF data set.

SYSDUMP
System dump statistics are to be written immediately to the SMF data set.

TABLEMGR
Table manager statistics are to be written immediately to the SMF data set.

TCLASS
Transaction class statistics are to be written immediately to the SMF data set.

TCPIP
Request global statistics for IP sockets.

TCPIPSERVICE
TCP/IP service statistics are to be written immediately to the SMF data set.

TDQUEUE
Transient data statistics are to be written immediately to the SMF data set.

TERMINAL
Terminal statistics are to be written immediately to the SMF data set.

TRANSACTION
Transaction statistics are to be written immediately to the SMF data set.

TRDUMP
Transaction dump statistics are to be written immediately to the SMF data set.

TSQUEUE
Temporary storage statistics are to be written immediately to the SMF data set.

URIMAP
URIMAP statistics are to be written immediately to the SMF data set.

VTAM
VTAM is now z/OS Communications Server. z/OS Communications Server statistics are to be written immediately to the SMF data set.

WEBSERVICE
WEBSERVICE statistics are to be written immediately to the SMF data set.

XMLTRANSFORM
XMLTRANSFORM statistics are to be written immediately to the SMF data set.

CEMT SET commands

The SET command changes the status of one or more named instances of a resource (such as a terminal), a particular subgroup of resources (such as a class of terminals), or all resources of a particular type (such as all terminals).

The SET command makes the changes that you specify, and displays the new status of the resources. No changes are made if there are syntax errors in the SET command.

To change the attributes of a resource, enter SET (or a suitable abbreviation) on the command line. The keywords you can use with CEMT SET are described in the rest of this section.

On every SET command, an argument, a generic argument, or ALL must be specified. For example:

```
SET TERMINAL(TRM1) INSERVICE
```

sets terminal TRM1 in service.

```
SET TERMINAL(TRM1 TRM2 TRM3) INSERVICE
```

or

```
SET TERMINAL(TRM1,TRM2,TRM3) INSERVICE
```

sets terminals TRM1, TRM2, and TRM3 in service.

```
SET TERMINAL(TRM*) INSERVICE
```

sets all terminals with identifiers starting with TRM in service.

```
SET TERMINAL ALL INSERVICE
```

sets all terminals in service.

However,

```
SET TERMINAL INSERVICE
```

produces an error.

CEMT SET ATOMSERVICE

Enable or disable an ATOMSERVICE resource definition.

In the CICS Explorer, the Atom services operations view provides a functional equivalent to this command.

If the optional attributes are omitted, the command has no effect.

If you disable an ATOMSERVICE resource definition, CICS returns an HTTP response to the Web client with a 503 (Service Unavailable) status code.

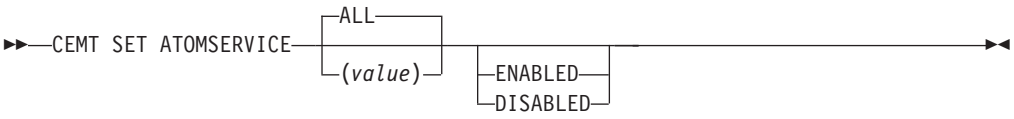
Syntax

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET ATOMSERVICE` (or suitable abbreviations for the keywords), followed by one or more ATOMSERVICE identifiers, or ALL. The resulting display lists the current status, similar to that obtained by using the **CEMT INQUIRE ATOMSERVICE** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET ATOMSERVICE` (or suitable abbreviations for the keywords), followed by one or more ATOMSERVICE identifiers or ALL, then followed by the attribute setting that you want to change. For example, `cemt s ato al e` resets the values for all ATOMSERVICE resource definitions to make them available for use (enabled).

Typing `?` at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

SET ATOMSERVICE



Conditions: NOTAUTH, NOTFND

Options

ALL

Any changes you request are made to all resources of the specified type that you are authorized to access.

(*value***)**
Specifies the 1- to 8-character name of the ATOMSERVICE resource definition.

ENABLESTATUS
Sets the status of the ATOMSERVICE resource definition:

ENABLED
The ATOMSERVICE resource definition is ready for use.

DISABLED
The ATOMSERVICE resource definition is disabled. An ATOMSERVICE resource definition in this state can be discarded.

CEMT SET AUTOINSTALL

Change attributes that control the automatic installation (autoinstall) of z/OS Communications Server terminals, APPC connections, virtual terminals (bridge facilities) used by the 3270 bridge mechanism, and MVS consoles.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

You can use the **CEMT SET AUTOINSTALL** command to change values that control autoinstall. For information about autoinstall, see the *CICS Resource Definition Guide*.

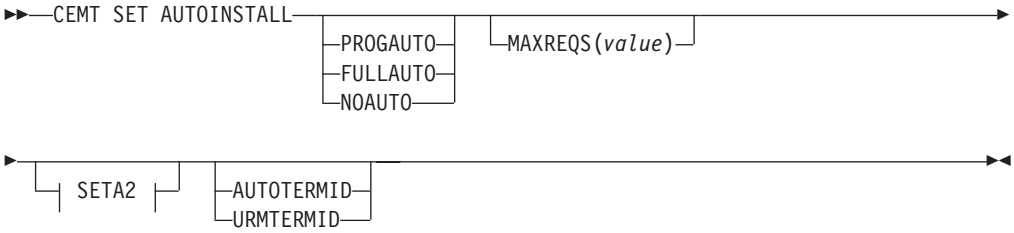
Syntax

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET AUTOINSTALL` (or suitable abbreviations for the keywords). The resulting display lists the current status, similar to that obtained by using the **CEMT INQUIRE AUTOINSTALL** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET AUTOINSTALL` (or suitable abbreviations for the keywords), followed by one or more attribute settings that you want to change. For example, `CEMT S AUT M(NEWVALUE P(NEWNAME)` resets the values for maximum number of requests for a new program.

Typing `?` at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET AUTOINSTALL



SETA2

```
├─PROGRAM(value)──────────────────────────────────────────────────────────┤
```

Options

AIBRIDGE
Specifies whether the autoinstall user replaceable module (URM) is to be called for bridge facilities. The values are as follows:

AUTOTERMID
Bridge facilities are to be defined automatically by CICS. The autoinstall URM is not to be called.

URMTERMID
The autoinstall URM is to be called.

CONSOLES
Specifies whether CICS autoinstalls an MVS console device when it receives an MVS MODIFY command from a console that is not defined.

PROGAUTO
MVS consoles are autoinstalled, and CICS calls the user autoinstall control program to obtain the termid and other user-specified information.

FULLAUTO
MVS consoles are autoinstalled by CICS automatically, without calling the user autoinstall control program. CICS assigns the termid for the console automatically, using the ¬ (logical not) symbol as the first character.

NOAUTO
Autoinstall for consoles is not allowed.

MAXREQS (*value*)
Specifies the largest number of autoinstall requests that are allowed to queue at one time, in the range 0 - 999.

You can prevent more terminals from logging on through autoinstall by setting this value to 0. This allows autoinstalled entries for terminals currently logged on to be deleted by the autoinstall program when they log off.

PROGRAM (*value*)
Specifies the name of the user program that controls the autoinstall process. A default program is supplied.

CEMT SET AUXTRACE

Change auxiliary tracing options.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

For more information about traces, see the *CICS Problem Determination Guide*. You might also find it useful to read the description of the CETR transaction.

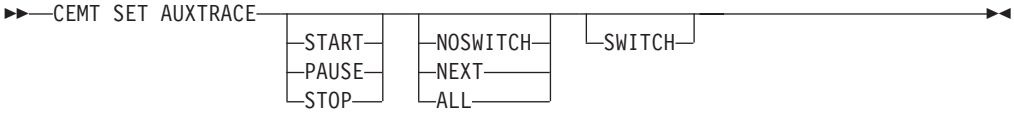
Syntax

Press the Clear key to clear the screen. There are two ways of starting this transaction:

- Type `CEMT SET AUXTRACE` (or suitable abbreviations for the keywords). The resulting display lists the current status, similar to that obtained by using the **`CEMT INQUIRE AUXTRACE`** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET AUXTRACE` (or suitable abbreviations for the keywords), followed by one or more attribute settings that you want to change. For example, `cemt s aux sto a` stops auxiliary tracing and causes automatic switching to occur as necessary.

Typing `?` at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET AUXTRACE



Options

ALL
Specifies that automatic switching between the two auxiliary trace data sets occurs as necessary until the end of this CICS session, without the need for operator intervention.

NEXT
Automatic switching of the auxiliary trace data set occurs when the current trace data set is full, but not subsequently.

NOSWITCH
Switching between the auxiliary trace data sets does not occur without operator intervention.

PAUSE
Auxiliary tracing stops, but the data set remains open. A subsequent START request causes trace entries to be written immediately following those that were written before the PAUSE request.

START
CICS starts auxiliary tracing and opens the auxiliary trace data set if it is currently closed.

STOP
CICS stops auxiliary tracing and closes the auxiliary trace data set. A subsequent START request causes new trace entries to be written at the start of the data set, thereby overwriting the trace entries that were written before the STOP request.

SWITCH
Specifies that an immediate switch from the current auxiliary trace data set is to be made to the alternate trace data set.

CEMT SET BRFACILITY

Release virtual terminals (bridge facilities) used by the 3270 bridge mechanism.

Description

You can use the **CEMT SET BRFACILITY** command to request deletion of the virtual terminal (bridge facility) used by the 3270 bridge mechanism.

If a transaction is currently running, the bridge facility is deleted at the end of the transaction. If the bridge facility is currently AVAILABLE, the facility is deleted at the next garbage clearance.

When a bridge facility is released, the delete function of the XFAINTU global user exit is driven.

Note: Bridge facilities are deleted only in the region in which the command is issued. Bridge facilities can exist in both router and AOR regions. This command deletes the facility in the region on which it is issued. It does not affect the other region, but this means that the bridge facility can no longer be used. However, to free the storage that a bridge facility occupies, you must issue this command in both regions. This command can be issued only in the router or AOR region where the bridge facility was created.

Syntax

Press the Clear key to clear the screen. There are two ways to start this transaction:

- Type `CEMT SET BRFACILITY` (or suitable abbreviations for the keywords), followed by (*value*) or ALL. The resulting display lists the current status, similar to that obtained by using the **CEMT INQUIRE BRFACILITY** command. You can tab to the highlighted field and overtype it with the required value.
- Type `CEMT SET BRFACILITY` (`CEMT S BR`) (or suitable abbreviations for the keywords), followed by (*value*) or ALL, then followed by one or more attribute settings that you want to change. For example, `cemt s br(value) rel` releases the specified bridge facility.

To obtain a syntax prompt, enter `?CEMT S BR`.

CEMT SET BRFACILITY

Options

(*value***)**
An 8-byte bridge facility token.

ALL
Any changes you request are made to all resources of the specified type that you are authorized to access.

RELEASED
Specifies that the bridge facility is marked for deletion.

CEMT SET BUNDLE

Enable or disable a BUNDLE resource.

In the CICS Explorer, the Bundles operations view provides a functional equivalent to this command.

If the optional attributes are omitted, the command has no effect.

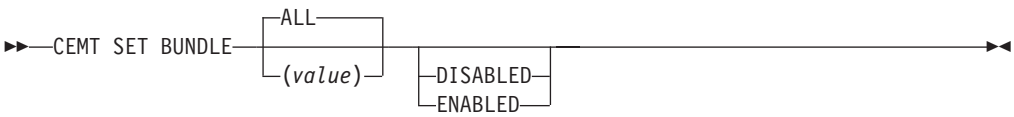
Syntax

Press the Clear key to clear the screen, You can start this transaction in two ways:

- Type `CEMT SET BUNDLE` followed by one or more BUNDLE identifiers or ALL. CEMT displays a list of BUNDLE resources and the current status of each resource. You can then tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET BUNDLE` followed by one or more BUNDLE identifiers or ALL, followed by the attribute setting that you want to change.

Typing `?` at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET BUNDLE



Options

ALL
Apply any changes to all of the specified resources that you are authorized to access.

DISABLED
Disable one or more BUNDLE resources. If you disable a BUNDLE resource definition, CICS tries to disable all of the resources that were created and enabled as part of the bundle deployment.

ENABLED
Enable one or more BUNDLE resources. You can enable a BUNDLE resource that is in any state except DISABLING or DISCARDING.

value
Specify the 1-8 character name of the BUNDLE resource definition.

CEMT SET CLASSCACHE

Set the status of autostart for the shared class cache.

In the CICS Explorer, the JVM classcache operations view provides a functional equivalent to this command.

Description

You can use the **SET CLASSCACHE** command to enable and disable the autostart for the class cache that is used by pooled JVMs in the CICS region. You cannot use this command on class caches that are used by JVM servers.

On an initial or cold start of CICS, the status of autostart is taken from the JVMCCSTART system initialization parameter. A value of YES or AUTO gives an autostart status of ENABLED and a value of NO gives an autostart status of DISABLED.

When you change the status of autostart for the shared class cache while CICS is running, subsequent warm or emergency CICS restarts use the changed setting. However, the shared class cache normally persists across warm or emergency restarts, so the changed setting normally has no effect in this situation. If the system is initial or cold started, the status of autostart is always taken from the **JVMCCSTART** system initialization parameter.

Syntax

CEMT SET CLASSCACHE

Options

DISABLED
Sets the status of autostart for the shared class cache to Disabled.

When autostart is disabled, if the shared class cache has been stopped or has not yet been started on this CICS run, an explicit **PERFORM CLASSCACHE** command is required to start it. If the status of the shared class cache is STOPPED and autostart is disabled, and CICS receives a request to run a Java application in a JVM with a profile that requires the use of the shared class cache, the application cannot run.

ENABLED
Sets the status of autostart for the shared class cache to Enabled.

When autostart is enabled, if the shared class cache has been stopped or has not yet been started on this CICS run, the shared class cache is started as soon as CICS receives a request to run a Java application in a JVM with a profile that requires the use of the shared class cache.

CEMT SET CONNECTION

Change the attributes of an MRO or ISC over SNA connection.

In the CICS Explorer, the ISC/MRO connections operations view provides a functional equivalent to this command.

Description

You can use the **CEMT SET CONNECTION** command to change some attributes that define an MRO or ISC over SNA connection.

Note: See also "CEMT SET IPCONN" on page 473. The **CEMT SET IPCONN** command is used to change the attributes of IPIC connections (also known as "IPCONNs").

For information about the different kinds of intercommunication connections, see the *CICS Intercommunication Guide*.

When a connection has been defined as INDIRECT, no information is given unless the direct connection to which it refers is installed.

To install a new connection definition when one is already installed, you must set the connection OUTSERVICE and, if it is z/OS Communications Server-connected, release it before using the CEDA INSTALL commands for your new definition.

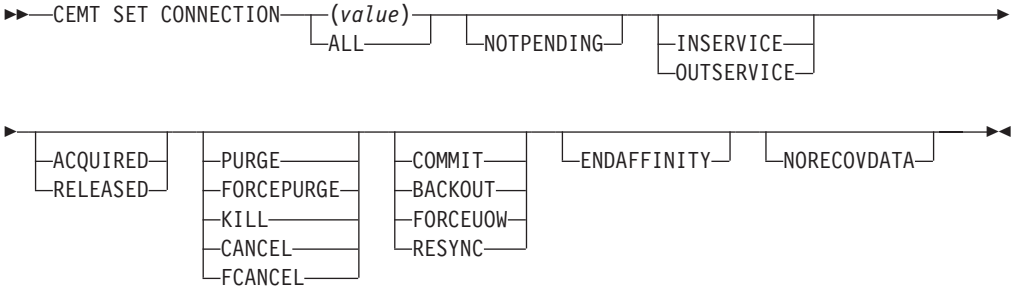
Syntax

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET CONNECTION` (or suitable abbreviations for the keywords), followed by one or more connection identifiers, or ALL. The resulting display lists the current status, similar to that obtained by using the **CEMT INQUIRE CONNECTION** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET CONNECTION` (or suitable abbreviations for the keywords), followed by one or more connection identifiers or ALL, followed in turn by one or more attribute settings that you want to change. For example, `cemt s c al i` resets the values for all connections to make them available for use (inservice).

Typing `?` at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET CONNECTION



Options

(*value*)
One or more names (1 - 4 characters) defined for an interregion communication (IRC) or an intersystem communication (ISC) connection.

ACQUIRED (APPC only)
Specifies whether CICS is to acquire a session with the logical unit represented by the CONNECTION name. To get more detailed information about the availability status of the connection elements, use the CEMT INQUIRE MODENAME command. A connection cannot be both ACQUIRED and OUTSERVICE.

ALL
Any changes you request are made to all resources of the specified type that you are authorized to access.

BACKOUT (APPC PARALLEL-SESSION, CICS-TO-CICS MRO, AND LU61 ONLY)
All UOWs shunted because of the failure of this connection are to be backed out. The normal resynchronization process is to be partially overridden: decisions are taken for any units of work that are indoubt because of a failure of the connection, but the decisions are recorded and any data inconsistencies are reported when the connection is next acquired.

CANCEL
Automatic initiate descriptors (AIDs) queuing for the specified connection are to be canceled.

AIDs representing scheduled and allocated requests waiting in the local CICS system for the specified connection are canceled. However, TD AIDs with an associated triggered task already started are not be canceled. In addition, the following CICS system AIDs are not purged unless FCANCEL is specified.

Table 7. System AIDs requiring FCANCEL to remove them

Description	Tranid
Remote delete AIDs	
Remote scheduler AIDs	CRSR
LU6.2 service manager 1 AIDs	CLS1
LU6.2 service manager 3 AIDs	CLS3
Remote scheduler PURGE AIDs	CRSQ
Resource manager resync AIDs	CRSY
Autoinstalled terminal delete AIDs	CATD
Autoinstalled terminal restart AIDs	CATR

When a canceled SCHEDULE request is found to have a precursor in a remote CICS system; that is, the AID was originally scheduled in a remote system, the remote AID is canceled asynchronously.

Message DFHTF0100 is written to CSMT to indicate how many AIDs have been deleted for the connection and how many remain.

An 'AIDS CANCELED' message appears on the CEMT panel whenever AIDs are deleted using the CANCEL option of the CEMT SET CONNECTION command.

COMMIT (APPC parallel-session, CICS-to-CICS MRO, and LU61 only)
All UOWs shunted because of the failure of this connection are to be committed. The normal resynchronization process is to be partially overridden: decisions are taken for any units of work that are indoubt because of a failure of the connection, but the decisions are recorded and any data inconsistencies are reported when the connection is next acquired.

ENDAFFINITY (APPC and LU6.1 only)
Specifies, where CICS is a member of a z/OS Communications Server generic resource group, that z/OS Communications Server is to end an affinity owned by CICS. This option is valid only for APPC and LU6.1 connections. The connection must be out of service and, for APPC, in NORECOVDATA state.

Note:

1. There is no facility in z/OS Communications Server for inquiring on affinities, so CICS has no certain knowledge that an affinity exists for a given connection. Whenever there is a possibility that an affinity has been created that you must end explicitly, CICS issues message DFHZC0177. This message gives the NETNAME of the suspect connection.
2. If a request to end an affinity is rejected by z/OS Communications Server because no such affinity exists, CICS issues message DFHZC0181.
3. A response of 'NETID 0 USE PRFRM' means that you cannot use SET CONNECTION to end the affinity because the value of NETID in the installed connection is 0. (NETID is the name by which the network containing the connected LU is known to z/OS Communications Server.) To end the affinity, you must use the PERFORM ENDAFFINITY command, on which you must specify the correct NETID.

   This response can also result from repeatedly hitting the ENTER key after the affinity has been ended.
4. Generic resources and affinities are described in the *CICS Intercommunication Guide*.

FCANCEL
All AIDs, including system AIDs, queuing for the specified connection are to be canceled. See Table 7 on page 434 for a list of those system AIDS that require FCANCEL to remove them. This can lead to unpredictable results and should be used only in exceptional circumstances.

Note: FCANCEL does not remove transient data AIDs with an associated triggered task. These aids may be removed by purging the associated task.

An 'AIDS CANCELED' message appears on the CEMT panel whenever AIDs are deleted using the FCANCEL option of the CEMT SET CONNECTION command.

FORCEPURGE (z/OS Communications Server only)
All transactions running on sessions on the connected system are immediately terminated abnormally. This can lead to unpredictable results and should be used only in exceptional circumstances.

In some extreme cases (for example, if an error occurs during backout processing), CICS might terminate abnormally.

FORCEPURGE replaces PURGE FORCE which is retained only for compatibility purposes. You should use FORCEPURGE in new applications.

FORCEUOW (APPC PARALLEL-SESSION, CICS-TO-CICS MRO, AND LU61 ONLY)
All UOWs shunted because of the failure of this connection are to be forced to back out or commit, as specified in the ACTION option of the TRANSACTION definition. The normal resynchronization process is to be partially overridden: decisions are taken for any units of work that are indoubt because of a failure of the connection, but the decisions are recorded and any data inconsistencies are reported when the connection is next acquired.

INSERVICE
The system is in service; that is, it is available for use.

For an MRO connection, all sessions are placed in service and the following occurs:

- If both the issuing system and system sysid have IRC open, and the issuing system has status INSERVICE within system sysid, connections are established and made available between the two systems.

- Otherwise, the status of the system (and the status of the underlying sessions) is set INSERVICE, so that when both systems have IRC open and are INSERVICE with respect to each other, connections are established. Note that INQUIRE indicates that systems (and their underlying sessions) are INSERVICE even though no connections can be used.
- The status of the underlying sessions for a system is always the same as that for the system itself.

For an ISC system, the following occurs:

- LU6.1 system (that is, links to sysid are through intersystem communication): all underlying sessions are placed in service.
- APPC system: causes the SNASVCMG sessions to be placed in service thereby enabling the connection subsequently to be established.

KILL
Terminate the task. System and data integrity is not guaranteed. The KILL option extends the PURGE and FORCEPURGE options. Use this option only after first attempting a purge or forced purge. The KILL option does not guarantee integrity of any kind, but in some situations you can use it to free up a stalled region, thus enabling the region to continue processing. In some cases, for example, if a task is killed during backout processing, CICS terminates abnormally.

NORECOVDATA (APPC only)
Specifies that the normal resynchronization process is to be overridden. NORECOVDATA forces indoubt units of work (according to the transaction definitions), targets any resyncs that were outstanding for the connection, and erases the logname previously received from the partner system. The state of the connection is reset.

Note: You should use SET CONNECTION NORECOVDATA only in exceptional circumstances. It erases recovery information and may compromise data integrity for units of work that have updated resources on remote systems.

Examples of circumstances in which you might need to use it are:

- You need to discard a connection, or issue a SET CONNECTION ENDAFFINITY command, and it is not possible for the quiesce protocols with the partner system to be completed. (Neither action is possible for an APPC connection if recovery data is outstanding.)
- An operational or logic error results in a logname mismatch for the connection. The connection state must be reset to allow the exchange lognames process to complete.

Note: If you specify NORECOVDATA you cannot specify COMMIT, BACKOUT, FORCEUOW, RESYNC, or NOTPENDING.

NOTPENDING (APPC and CICS-to-CICS MRO only)
Specifies, for either of the following kinds of connection, that the normal resynchronization process is to be overridden:

- A connection to a CICS TS for z/OS, Version 4.2 partner that has performed an initial start
- A connection to a pre- CICS TS for z/OS, Version 4.2 partner that has performed a cold start.

NOTPENDING forces all indoubt units of work (according to the transaction definition) that were created by the connection before the initial (or cold) start

of the partner. It also forgets any resyncs (waitforget UOW-links) that are outstanding for the connection, and created before the initial (or cold) start of the partner.

The PENDING condition indicates the existence of recovery information (either shunted UOWs or decisions remembered for the partner) on a connection that has experienced a lognames mismatch with its partner. For a CICS TS for z/OS, Version 4.2 partner, a lognames mismatch indicates that the partner has performed an initial start. For a pre-CICS TS for z/OS, Version 4.2 partner, a lognames mismatch indicates that the partner has performed a cold start. In either case, the recovery protocol has been corrupted by a loss of log data at the partner.

It is not possible to set a connection to NOTPENDING state (forcing indoubt and erasing NOFORGET UOWs) until this system has made contact with the partner and received a new logname from it.

Decisions for a whole connection can be forgotten, but that does not affect the memory of a decision for any other connection involved in the UOW.

Note:

1. If you specify NOTPENDING you cannot specify COMMIT, BACKOUT, FORCEUOW, RESYNC, or NORECOVDATA. For advice on which option to use, see CEMT SET CONNECTION - notes on RES ync.
2. NOTPENDING has no effect on MRO connections to pre-CICS TS for z/OS, Version 4.2 systems. However, the cold start of a pre-CICS TS for z/OS, Version 4.2 MRO partner causes the SET CONNECTION NOTPENDING function to be performed automatically, session by session.

The exchange lognames function and the resynchronization function are described in the *CICS Intercommunication Guide* and the *Systems Network Architecture—LU6.2 Reference: Peer Protocols* manual.

OUTSERVICE

The system is out of service; that is, it is not available for use.

For an MRO connection, all sessions are placed out of service (immediately if PURGE is specified, or when tasks have terminated if it is not) and the following occurs:

- If the connection is currently ACQUIRED, the sessions are broken (quiesced). The connection cannot be used until it is once again placed INSERVICE.
- If the connection is currently RELEASED, the status of the connection is set OUTSERVICE and it cannot be used until it is INSERVICE again.
- The status of the underlying sessions for a connection is always the same as that for the connection itself.

For an ISC system, the following occurs:

- LU6.1 system—all underlying sessions owned by the system are released and placed out of service: immediately if PURGE or FORCEPURGE is specified; or when tasks have terminated if neither PURGE nor FORCEPURGE is specified.

  If the response to an INQUIRE CONNECTION command shows OUTSERVICE, it does not imply that the connection has been explicitly set as SET OUTSERVICE; in particular circumstances, you cannot reinstall this connection.
- APPC system—this option is valid only if the system is released. All sessions owned by the ISC system are then out of service and released.

PURGE (z/OS Communications Server only and IRC only)
Transactions running on the connected system are abnormally terminated. Transactions are terminated only if system and data integrity can be maintained. A transaction is not purged if its definition Specifies SPURGE=NO.

RELEASED (APPC only)
Specifies whether CICS is to release a session with the logical unit represented by the CONNECTION name. To get more detailed information about the availability status of the connection elements, use the CEMT INQUIRE MODENAME command.

RESYNC (*MRO TO CICS TS for z/OS, Version 4.2 AND LATER SYSTEMS AND APPC ONLY***) (APPC PARALLEL-SESSION, CICS-TO-CICS MRO, AND LU61 ONLY)**
Any UOWs shunted because of the failure of this connection are to be retried (that is, exchange lognames resynchronization for this connection is to be attempted). This process should normally be started automatically when a connection is acquired or when a UOW is unshunted. The normal resynchronization process is to be partially overridden: decisions are taken for any units of work that are indoubt because of a failure of the connection, but the decisions are recorded and any data inconsistencies are reported when the connection is next acquired.

Note:

1. The COMMIT, BACKOUT, FORCEUOW and RESYNC operations are synchronous with setting the state of the UOW; that is, an INQUIRE UOW following SET CONNECTION BACKOUT, COMMIT, FORCEUOW, or RESYNC returns the new UOW states.
2. Specifying one of these options unshunts all units of work that have failed because of a failure of the connection. Before issuing SET CONNECTION FORCEUOW, you may want to use the SET UOW command to specify commit or backout for each indoubt unit of work explicitly, rather than letting it default, Local procedures determine the importance of the data and the method of using the INQUIRE UOW, INQUIRE UOWENQ, and INQUIRE UOWLINK commands to establish the correct actions.
3. You can specify only one of the BACKOUT, COMMIT, FORCEUOW, RESYNC, NOTPENDING, and NORECOVDATA options. SET CONNECTION NORECOVDATA should be used only in exceptional circumstances.
4. To force all indoubt units of work caused by a failure of the connection in the same direction, use SET CONNECTION COMMIT or SET CONNECTION BACKOUT.
5. The BACKOUT, COMMIT, FORCEUOW, or RESYNC options of SET CONNECTION and SET UOW do not clear resync information. If you want to do this, you must use SET CONNECTION NOTPENDING or SET CONNECTION NORECOVDATA.
6. You can issue BACKOUT, COMMIT, FORCEUOW, or RESYNC commands *before* issuing SET CONNECTION NOTPENDING or SET CONNECTION NORECOVDATA.

CEMT SET CORBASERVER

Enable or disable a CorbaServer, turn its automatic publishing feature on or off, and set the session bean timeout value.

Description

If the optional attributes are omitted, the command has no effect.

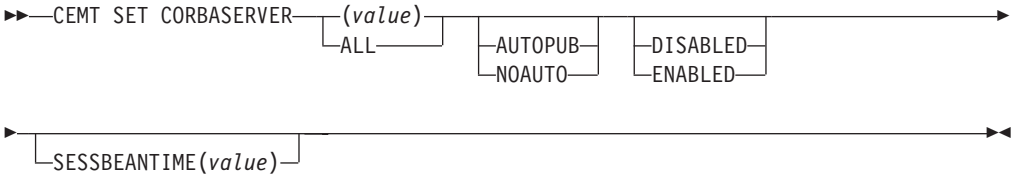
Syntax

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET CORBASERVER` (or suitable abbreviations for the keywords), followed by one or more CORBASERVER identifiers, or ALL. The resulting display lists the current status, similar to that obtained by using the **CEMT INQUIRE CORBASERVER** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET CORBASERVER` (or suitable abbreviations for the keywords), followed by one or more CORBASERVER identifiers or ALL, then followed by one or more attribute settings that you want to change. For example, `cemt s c al i` resets the values for all CORBASERVERs to make them available for use (inservice).

Typing `?` at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET CORBSERVER



Options

ALL
Any changes you request are made to all resources of the specified type that you are authorized to access.

(*value***)**
Specifies the 1-to 4-character name of the CorbaServer.

AUTOPUB
Specifies that the contents of a deployed JAR file are automatically published to the namespace when the DJAR definition is successfully installed into this CorbaServer.

Changing the setting of AUTOPUBLISH affects only DJAR definitions installed after the SET CORBASERVER command is executed. It has no effect on previously-installed DJAR definitions.

DISABLED
Specifies that the CorbaServer is disabled.

In a CICS region containing multiple CorbaServers, putting one of the CorbaServers into disabled state allows you to apply maintenance to it without bringing down CICS; the other CorbaServers remain available.

Similarly, in a logical EJB/CORBA server containing multiple AORs (each containing multiple CorbaServers), putting a CorbaServer on one of the AORs into disabled state allows you to apply maintenance to the CorbaServer without needing to remove the AOR from the routing set.

You can disable a CorbaServer that is in any state except ENABLING or DISCARDING. When you do so, any currently-executing methods are allowed to continue to completion. Until all its objects have been destroyed or passivated, the CorbaServer enters DISABLING state. Note that:

- Any instance of an enterprise bean or CORBA stateless object that is not participating in an OTS transaction is destroyed or passivated at the end of the currently-executing method.
- Any instance of an enterprise bean or CORBA stateless object that *is* participating in an OTS transaction is not destroyed or passivated until the end of the OTS transaction; typically, any future method calls against this instance (within the scope of the OTS transaction) will succeed. At the end of the OTS transaction, the instance is destroyed or passivated.

If the workload manager directs new work to an AOR in which the target CorbaServer is in DISABLING or DISABLED state, it is informed that the CorbaServer is disabled.

Note: "New work" means:

- Requests for methods that will run under a new OTS transaction. (A "new OTS transaction" means an OTS transaction in which none of the AORs in the logical server is already participating, before the method call; *not* necessarily an OTS transaction that is started immediately before or after the method call.)
- Requests for methods that will run under no OTS transaction.

ENABLED
Specifies that the CorbaServer is enabled. You can enable a CorbaServer that is in any state except DISABLING or DISCARDING.

NOAUTO
Specifies that the contents of a deployed JAR file are not automatically published to the namespace when the DJAR definition is successfully installed into this CorbaServer.

SESSBEANTIME (*value***)**
Specifies the elapsed time period of inactivity, after which a session bean can be discarded. The time period is in minutes, in the range 0 through 14399. 9A value of 0 indicates that beans are not timed out.

CEMT SET DB2CONN

Sets attributes of the CICS DB2 connection, including the pool and command thread attributes.

In the CICS Explorer, the DB2 connections operations view provides a functional equivalent to this command.

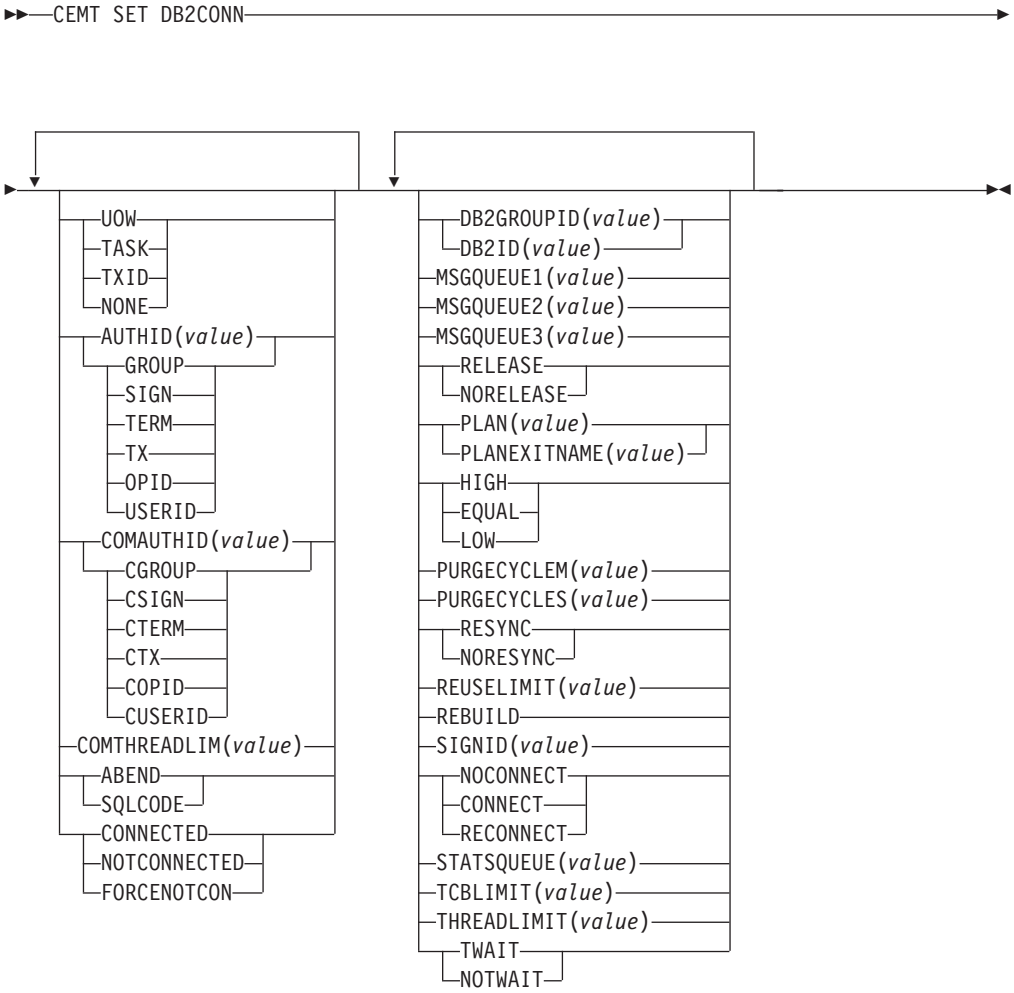
Syntax

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET DB2CONN` (or suitable abbreviations for the keywords). The current status is displayed, similar to that obtained by using the **CEMT INQUIRE DB2CONN** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET DB2CONN` (or suitable abbreviations for the keywords), followed by one or more attribute settings that you want to change.

Typing ? at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET DB2CONN



Options

ACCOUNTREC

Specifies the minimum amount of DB2 accounting required for transactions requiring pool threads. The specified minimum can be exceeded, as described in the following options.

UOW The CICS DB2 attachment facility causes an accounting record to be produced for each unit of work (UOW), assuming that the thread has been released at the end of the UOW.

TASK

The CICS DB2 attachment facility causes a minimum of one accounting record for each CICS task to be produced.

A transaction containing multiple UOWs can use a different thread for each of its UOWs (assuming that the thread is released at sync point). The result can be an accounting record produced for each UOW.

TXID The CICS DB2 attachment facility causes an accounting record to be produced when the transaction ID that is using the thread changes.

Because pool threads are typically used by a number of different transaction IDs, there is an increased chance that a transaction containing multiple UOWs will use a different thread for each UOW (assuming that the thread is released at sync point. In this case an accounting record can be produced for each UOW.

NONE
No accounting records are required for transactions using pool threads.

DB2 does, however, produce at least one accounting record for each thread when the thread is terminated. Additionally, authorization changes cause accounting records to be produced.

AUTHID
Specifies what ID should be used for security checking for pool threads. If AUTHID is specified, AUTHTYPE is cleared.

AUTHTYPE
Specifies the type of ID that can be used for pool threads. If AUTHTYPE is specified AUTHID is cleared.

GROUP
Specifies the 8-character user ID and the connected group name as the authorization ID. The following table shows how these two values are interpreted by DB2.

IDs passed to DB2	How DB2 interprets values
CICS sign-on user ID (USERID)	Represents the primary DB2 authorization ID.
RACF connected group name	If the RACF list of group options is not active, then DB2 uses the connected group name supplied by the CICS attachment facility as the secondary DB2 authorization ID. If the RACF list of group options is active, DB2 ignores the connected group name supplied by the CICS attachment facility, but the value appears in the DB2 list of secondary DB2 authorization IDs.

To use the GROUP option the CICS system must have SEC=YES specified in the CICS system initialization table (SIT).

If no RACF group ID is available for this user ID, an 8-character field of blanks is passed to DB2 as the group ID.

SIGN Specifies the SIGNID parameter of the DB2CONN should be used as the resource authorization ID.

TERM
Specifies the terminal identification (four characters padded to eight) as an authorization ID. An authorization ID cannot be obtained in this manner if a terminal is not connected with the transaction.

If a transaction is started (using a CICS command) and has no terminal associated with it, AUTHTYPE(TERM) should not be used.

TX Specifies the transaction identification (four characters padded to eight) as the authorization ID.

OPID The operator identification associated with the user ID that is associated with the CICS transaction is used as the authorization ID (three characters padded to eight).

USERID
The 8-character user ID associated with the CICS transaction is used as the authorization ID.

When the DB2 sample sign-on exit DSN3@SGN is used with AUTHTYPE(USERID), the exit sends the user ID to DB2 as the primary authorization ID and the RACF group ID to DB2 as the secondary ID. When the sample sign-on exit is used, there is no difference between AUTHTYPE(USERID) and AUTHTYPE(GROUP).

COMAUTHID
Specifies which ID should be used for security checking when using command threads. If COMAUTHID is specified, COMAUTHTYPE is cleared.

COMAUTHTYPE
Specifies the type of ID that can be used for security checking when using command threads. If COMAUTHTYPE is specified, COMAUTHID is cleared.

CGROUP
Specifies the 8-character user ID and the connected group name as the authorization ID. The following table shows how these two values are interpreted by DB2.

IDs passed to DB2	How DB2 interprets values
CICS sign-on user ID (USERID)	Represents the primary DB2 authorization ID.
RACF connected group name	If the RACF list of group options is not active, then DB2 uses the connected group name supplied by the CICS attachment facility as the secondary DB2 authorization ID. If the RACF list of group options is active, DB2 ignores the connected group name supplied by the CICS attachment facility, but the value appears in the DB2 list of secondary DB2 authorization IDs.

To use the CGROUP option the CICS system must have SEC=YES specified in the CICS system initialization table (SIT).

If no RACF group ID is available for this user ID, an 8-character field of blanks is passed to DB2 as the group ID.

CSIGN
Specifies the SIGNID parameter of the DB2CONN should be used as the resource authorization ID.

CTERM
Specifies the terminal identification (four characters padded to eight) as an authorization ID. An authorization ID cannot be obtained in this manner if a terminal is not connected with the transaction.

If a transaction is started (using a CICS command) and has no terminal associated with it, the COMAUTHTYPE(TERM) should not be used.

CTX Specifies the transaction identification (four characters padded to eight) as the authorization ID.

COPID
The operator identification associated with the user ID that is

associated with the CICS transaction sign-on facility is used as the authorization ID (three characters padded to eight).

CUSERID
The 8-character user ID associated with the CICS transaction is used as the authorization ID.

When the DB2 sample sign-on exit DSN3@SGN is used with AUTHTYPE(USERID), the exit sends the user ID to DB2 as the primary authorization ID and the RACF group ID to DB2 as the secondary ID. When the sample sign-on exit is used, there is no difference between COMAUTHTYPE(CUSERID) and COMAUTHTYPE(CGROUP).

COMTHREADLIM
Specifies the current maximum number of command threads that the CICS DB2 attachment allows to be active before requests overflow to the pool.

CONNECTERROR
Returns a value when CICS is not connected to DB2 because the adapter is in standby mode. The value shows how this information is to be passed back to an application that has issued a SQL request.

ABEND
The application is stopped with abend code AEY9.

SQLCODE
The application receives a -923 SQLCODE.

CONNECTST
Sets the status of the CICS DB2 connection; that is, to start or to stop the CICS DB2 connection.

CONNECTED
This is equivalent to issuing DSNC STRT to start the CICS DB2 attachment. If the requested DB2 subsystem is active, control returns when CICS and DB2 have been connected. If the requested DB2 subsystem is not active, the response returned is dependent on the setting of STANDBYMODE: If DB2 is not initialized, and STANDBYMODE(NOCONNECT) is specified on the DB2CONN, INVREQ and RESP2=39 is returned. If you specify STANDBYMODE(CONNECT) or STANDBYMODE(RECONNECT), NORMAL with RESP2=38 is returned indicating that the CICS DB2 attachment is in standby mode and will connect to DB2 as soon as it becomes active.

NOTCONNECTED
CEMT initiates a quiesce stop of the CICS DB2 connection and implicitly uses the NOWAIT option, meaning that control returns immediately to CEMT before the quiesce is complete.

FORCENOTCON
FORCENOTCON force stops the connection by force purging transactions currently using DB2. Control is not returned until the connection is stopped.

DB2GROUPID
Specifies the name (up to four characters) of a data-sharing group of DB2 subsystems. CICS attempts to connect to any active member of this group, using group attach. With DB2 Version 10, the name can be a subgroup name identifying a subset of the data-sharing group.

DB2GROUPID can only be changed when CICS is not connected to a DB2 system. Specifying a DB2GROUPID causes the DB2ID in the installed DB2CONN definition to be blanked out. If an individual subsystem's DB2ID is specified in a CEMT or EXEC CICS SET DB2CONN command, or in a DSNC STRT command, this overrides any DB2GROUPID that is set in the installed DB2CONN definition. The DB2GROUPID is blanked out, and needs to be set again (using CEDA or a SET DB2CONN command) to use group attach. Also note that you cannot set a DB2GROUPID and a DB2ID in the same command, this causes the command to fail.

DB2ID
Specifies the name of the DB2 subsystem that the CICS DB2 attachment should connect to. DB2ID can only be changed when CICS is not connected to a DB2 system. Specifying a DB2ID causes the DB2GROUPID in the installed DB2CONN definition to be blanked out, and the DB2GROUPID needs to be set again to use group attach. If a DB2GROUPID is specified in a CEMT or EXEC CICS SET DB2CONN command, this overrides any DB2ID that is set in the installed DB2CONN definition, and the DB2ID is blanked out. Also note that you cannot set a DB2ID and a DB2GROUPID in the same command, this causes the command to fail.

MSGQUEUE1
Specifies the first transient data destination to which unsolicited messages from the CICS DB2 attachment are sent.

MSGQUEUE2
Specifies the second transient data destination to which unsolicited messages from the CICS DB2 attachment are sent.

MSGQUEUE3
Specifies the third transient data destination to which unsolicited messages from the CICS DB2 attachment are sent.

NONTERMREL
Specifies whether non-terminal transactions release threads for reuse at intermediate sync points.

RELEASE
Non-terminal transactions release threads for reuse at intermediate sync points.

NORELEASE
Non-terminal transactions do not release threads for reuse at intermediate sync points.

PLAN
Specifies the name of the plan to be used for all threads in the pool. If PLAN is specified, PLANEXITNAME is cleared.

PLANEXITNAME
Specifies the name of the dynamic plan exit to be used for pool threads. If you change the PLAN and PLANExitname while there are active transactions for the pool, the next time the transaction releases the thread, the plan/exit will be determined using the new rules. If PLANExitname is specified, PLAN is cleared.

PRIORITY
Specifies the priority of the pool thread TCBs relative to the CICS main TCB (QR TCB). The thread TCBs are CICS open L8 TCBs.

HIGH
Thread TCBs have a higher priority than the CICS QR TCB.

EQUAL
Thread TCBs have equal priority with the CICS QR TCB.

LOW Thread TCBs have a lower priority than the CICS QR TCB.

PURGECYCLEM
Specifies the number of minutes (in the range 00 - 59) in the protected thread purge cycle time. Use this parameter in conjunction with PURGECYCLES.

PURGECYCLES
Specifies the number of seconds (in the range 00 - 59) in the protected thread purge cycle time. Use this parameter in conjunction with PURGECYCLEM. The minimum protected thread purge cycle time is 5 seconds and the default is 30 seconds.

A protected thread is not terminated immediately when it is released. It is terminated only after two completed purge cycles, if it has not been reused in the meantime. For example, if the protected thread purge cycle is set to 30 seconds, a protected thread is purged 30 - 60 seconds after that thread is released. An unprotected thread is terminated when it is released (at sync point or end of task) if there are no other transactions waiting for a thread on that DB2ENTRY.

RESYNCMEMBER
This applies only if you are using group attach, and specifies the strategy that CICS adopts if outstanding units of work are being held for the last DB2 data sharing group member to which CICS was connected. (Units of work which are shunted indoubt are not included in this process, because CICS itself is unable to resolve those units of work at this time. Resynchronization for those UOWs will occur when CICS has resynchronized with its remote coordinator.)

RESYNC
CICS connects to the same DB2 data sharing group member to resynchronize the outstanding units of work.

NORESYNC
CICS makes one attempt to connect to the same DB2 data sharing group member, and if that attempt fails, CICS connects to any member of the DB2 data sharing group and issues a warning about the outstanding units of work.

REUSELIMIT
Specifies a value in the range 0 - 10000 representing the maximum number of times a thread can be reused before it is terminated. The default is 1000. A value of 0 means that there is no limit on the number of times that a thread can be reused.

The reuse limit applies to unprotected threads both in the pool and on a DB2ENTRY, and to protected DB2ENTRY threads.

SECURITY
Specifies the security.

REBUILD
Specifies that the CICS DB2 attachment should force all existing threads to sign-on again at the next thread reuse. It should be used when RACF profiles have been updated by issuing the following commands:

- CEMT PERFORM SECURITY REBUILD for RACF 1.9.2 or earlier

- TSO SETROP TS RACLIST(xxxxxxxx) REFRESH for RACF 2.1 or later

SIGNID
Specifies the authorization ID to be used by the CICS DB2 attachment when signing on to DB2 for pool and DB2ENTRY threads specifying AUTHTYPE(SIGN), and command threads specifying COMAUTHTYPE(CSIGN).

STANDBYMODE
Specifies the action to be taken by the CICS DB2 attachment if DB2 is not active when an attempt is made to start the connection from CICS to DB2:

NOCONNECT
The CICS DB2 attachment should terminate.

CONNECT
The CICS DB2 attachment goes into standby mode to wait for DB2.

RECONNECT
The CICS DB2 attachment goes into standby mode and waits for DB2. Having connected to DB2, if DB2 then fails the CICS DB2 attachment reverts again to standby mode and later reconnects to DB2 when it comes up again.

STATSQUEUE
Specifies the transient data destination for CICS DB2 attachment statistics produced when the CICS DB2 attachment is shutdown.

TCBLIMIT
Specifies the maximum number of TCBs that can be used to process DB2 requests. When connected to DB2 Version 5 or earlier, the CICS DB2 attachment facility creates the TCBs in the form of subtasks up to the limit specified by TCBLIMIT. Each of these subtasks identifies to DB2 and creates a connection into DB2. When connected to DB2 Version 6 or later, CICS creates open TCBs (up to the limit specified in the system initialization parameter MAXOPENTCBS). The TCBLIMIT attribute of the DB2CONN definition governs how many of the open TCBs can be used to access DB2 — that is, how many of them can identify to DB2 and create a connection into DB2.

THREADLIMIT
Specifies the current maximum number of pool threads the CICS DB2 attachment allows active before requests are made to wait or are rejected according to the THREADWait parameter.

THREADWAIT
Specifies whether transactions should wait for a pool thread or be abended if the number of active pool threads reach the THREADlimit number.

The CICS DB2 attachment issues a unique abend code AD3T, message DFHDB2011 when THREADWAIT=NO is coded and the number of pool threads is exceeded.

TWAIT
If all threads are busy, a transaction must wait until one becomes available. A transaction can wait as long as CICS allows it to wait, generally until a thread becomes available.

NOTWAIT
If all threads are busy, the transaction is terminated with abend code AD3T or AD2P.

Note that when you change the value of AUTHId, AUTHType, COMAUTHId, COMAUTHType or SIgnid, a surrogate user security check is invoked if security is active. This ensures that the user ID under which SET is being executed is authorized to act on behalf of the user ID being set.

Note that when the SET DB2CONN command is specified all parameters, except the DB2ID (the connected subsystem), can be set when the CICS DB2 attachment is active. *DB2ID can only be changed by stopping and restarting the attachment*.

If you change the PLAN and PLANEXITNAME while there are active transactions for that entry, or the pool, the next time the transaction releases the thread, the plan/exit will be determined using the new rules.

CEMT SET DB2ENTRY

Sets the attributes of a particular DB2ENTRY resource used to define resources for a specific transaction or a group of transactions when accessing DB2.

In the CICS Explorer, the DB2 entries operations view provides a functional equivalent to this command.

Description

The entry is identified by the name it was identified with in CEDA.

All parameters on a **SET DB2ENTRY** command can be set while the CICS DB2 attachment is active and the transactions are active.

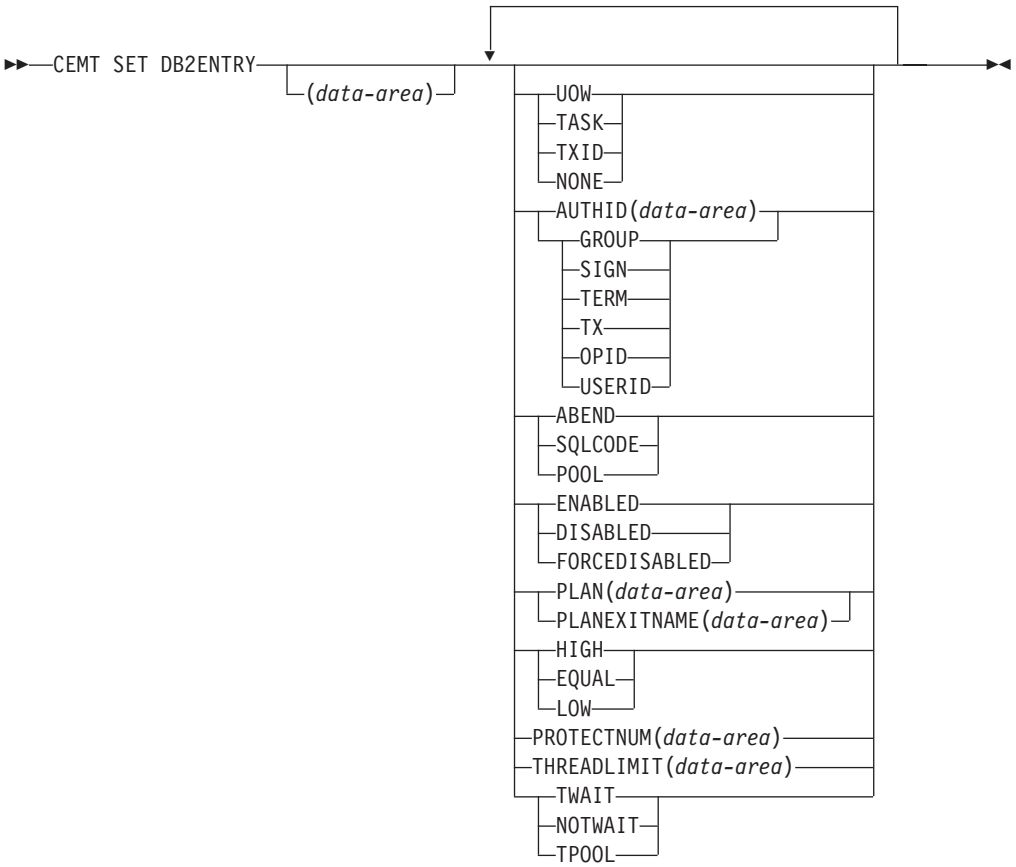
Syntax

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET DB2ENTRY` (or suitable abbreviations for the keywords), followed by either a value that corresponds to a DB2 entry name, or ALL. The resulting display lists the current status, similar to that obtained by using the **CEMT INQUIRE DB2ENTRY** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET DB2ENTRY` (or suitable abbreviations for the keywords), followed by either a value that corresponds to a DB2 entry name or ALL, then followed by one or more attribute settings that you want to change. For example, `cemt s db2e all ena` enables all DB2ENTRY resources.

Typing `?` at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET DB2ENTRY



Options

ACCOUNTREC

Specifies the minimum amount of DB2 accounting required for transactions using this DB2ENTRY resoource. The specified minimum can be exceeded, as described in the following options.

UOW The CICS DB2 attachment facility causes an accounting record to be produced for each unit of work (UOW), assuming that the thread is released at the end of the UOW.

TASK

The CICS DB2 attachment facility causes a minimum of one accounting record to be produced for each CICS task.

A transaction that contains multiple UOWs can use a different thread for each of its UOWs (assuming the thread is released at syncpoint). The result can be an accounting record produced for each UOW.

TXID The CICS DB2 attachment facility produces at least one accounting record to be produced when the transid using the thread changes.

This option applies to DB2ENTRY resources that are used by more than one transaction ID. As threads are typically released at syncpoint, a transaction that contains multiple UOWs can use a different thread for each UOW. The result can be that an accounting record is produced per UOW.

NONE
No accounting records are required for transactions using threads from this DB2ENTRY resource.

DB2 does, however, produce at least one accounting record for each thread when the thread is terminated. Additionally, authorization changes cause accounting records to be produced.

AUTHID
Specifies the id to use for security checking when using this DB2ENTRY resource. If AUTHID is specified, AUTHTYPE is cleared.

AUTHTYPE
Returns the type of id that can be used for security checking when using this DB2ENTRY resource. If AUTHTYPE is specified, AUTHID is cleared.

GROUP
Specifies the 8-character USERID and the connected group name as the authorization ID. The following table shows how these two values are interpreted by DB2.

IDs passed to DB2	How DB2 interprets values
CICS sign-on user ID (USERID)	Represents the primary DB2 authorization ID.
RACF connected group name	If the RACF list of group options is not active, DB2 uses the connected group name supplied by the CICS attachment facility as the secondary DB2 authorization ID. If the RACF list of group options is active, DB2 ignores the connected group name supplied by the CICS attachment facility, but the value appears in the DB2 list of secondary DB2 authorization IDs.

To use the GROUP option, the CICS system must have RACF external security SEC=YES specified in the CICS system initialization table (SIT).

If no RACF group ID is available for this USERID, an 8-character field of blanks is passed to DB2 as the group ID.

SIGN Specifies the SIGNID parameter of the DB2CONN should be used as the resource authorization ID.

TERM
Specifies the terminal identification (four characters padded to eight) as an authorization ID. An authorization ID cannot be obtained in this manner if a terminal is not connected with the transaction.

If a transaction is started (using a CICS command) and has no terminal associated with it, AUTHTYPE(TERM) should not be used.

TX Specifies the transaction identification (four characters padded to eight) as the authorization ID.

OPID The operator identification associated with the CICS transaction is used as the authorization ID (three characters padded to eight).

USERID
The 8-character USERID associated with the CICS transaction is used as the authorization ID.

When the DB2 sample sign-on exit DSN3@SGN is used with AUTHTYPE(USERID), the exit sends the user ID to DB2 as the primary

authorization ID and the RACF group ID to DB2 as the secondary ID. When the sample sign-on exit is used, there is no difference between AUTHTYPE(USERID) and AUTHTYPE(GROUP).

DISABLEDACT
Specifies the action for new transactions that access a DB2ENTRY resource when it has been disabled, or is disabling.

POOL
The CICS DB2 attachment facility routes the request to the pool. Message DFHDB2072 is sent to the transient data destination specified by MSGQUEUEn on the DB2CONN for each transaction routed to the pool.

ABEND
The CICS DB2 attachment facility abends the transaction. The abend code is AD26.

SQLCODE
An SQLCODE is returned to the application indicating that the DB2ENTRY resource is disabled.

ENABLEDSTATUS
Specifies whether the DB2ENTRY resource can be accessed by applications.

ENABLED
The DB2ENTRY resource can be accessed by applications.

DISABLED
The DB2ENTRY resource cannot be accessed by applications. A DB2ENTRY resource must be disabled before it can be reinstalled or discarded.

FORCEDISABLED
All tasks using the DB2ENTRY resource and those queued against the DB2ENTRY resource are force purged. The DB2ENTRY resource is then disabled.

PLAN
Specifies the name of the plan to be used for this DB2ENTRY resource.

If PLAN is specified, PLANEXITNAME is cleared.

PLANEXITNAME
Specifies the name of the dynamic plan exit to be used for this DB2ENTRY resource. If you change the PLAN and PLANEEXITNAME while there are active transactions for the DB2ENTRY resource, the next time the transaction releases the thread, the plan/exit is determined using the new rules. If PLANEXITNAME is specified, PLAN is cleared.

PRIORITY
Specifies the priority of the thread TCBs for this DB2ENTRY resource relative to the CICS main TCB (QR TCB). The thread TCBs are CICS open L8 TCBs.

HIGH
Thread TCBs have a higher priority than the CICS QR TCB.

EQUAL
Thread TCBs have equal priority with the CICS QR TCB.

LOW Thread TCBs have a lower priority than the CICS QR TCB.

PROTECTNUM
Specifies the maximum number of protected threads for this DB2ENTRY resource.

THREADLIMIT
Specifies the maximum number of threads for this DB2ENTRY resource that the CICS DB2 attachment allows active before requests are made to wait, are rejected, or are routed to the pool.

THREADWAIT
Specifies whether or not transactions should wait for a DB2ENTRY thread, be abended, or overflow to the pool if the number of active DB2ENTRY threads reach the THREADLIMIT number.

TWAIT
If all threads are busy, the transaction waits until one becomes available.

NOTWAIT
If all threads are busy, the transaction is terminated with an abend code AD2P.

TPOOL
If all threads are busy, the transaction is diverted to use the pool of threads. If the pool is also busy, and NOTWAIT has been specified for the THREADWAIT parameter on the DB2CONN. The transaction is terminated with abend code AD3T.

All parameters on **SET DB2ENTRY** can be set while the CICS DB2 attachment is active and the transactions are active.

CEMT SET DB2TRAN

Set the attributes of a specific DB2TRAN resource that is associated with a DB2ENTRY resource.

In the CICS Explorer, the DB2 transactions operations view provides a functional equivalent to this command.

Description

If a transaction identifer is specified on a DB2ENTRY resource when the DB2ENTRY is installed, CICS installs a DB2TRAN resource named DFH*xxxx*, where xxxx is the transaction identifer.

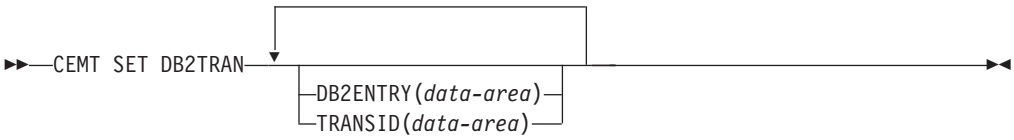
Syntax

Press the Clear key to clear the screen. There are two ways to start this transaction:

- Type `CEMT SET DB2TRAN` (or suitable abbreviations for the keywords), followed by either a value that corresponds to a DB2 transaction name, or ALL. You get a display that lists the current status, similar to that obtained by CEMT INQUIRE DB2TRAN. You can then tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET DB2TRAN` (or suitable abbreviations for the keywords), followed by either a value that corresponds to a DB2 transaction name, or ALL, then followed by one or more attribute settings that you want to change. For example, `cemt s db2t all db2e(entry1)` sets all DB2TRAN resources to refer to a DB2ENTRY resource named entry1.

Typing ? at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET DB2TRAN



Options

DB2ENTRY
Specifies the name of the DB2ENTRY to which this DB2TRAN refers; that is, the DB2ENTRY with which to associate this additional transaction identifer.

TRANSID
Specifies the transaction identifer to be associated with the entry. The transaction identifer can include wildcard characters (see the *CICS Resource Definition Guide* for information about use of wildcard characters). If you change TRANSID for a DB2TRAN resource while the attachment is active, all transactions with a thread continue to use the thread from that entry until it is released for reuse. When that transaction issues the next SQL statement, the thread is acquired from the entry or pool based upon the new definition.

You can set DB2TRAN parameters at any time.

CEMT SET DELETSHIPPED

Change system settings that control the CICS timeout delete mechanism.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Syntax

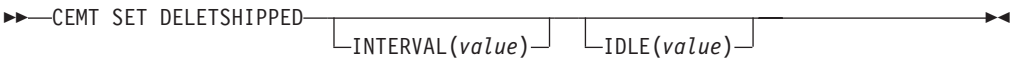
The CICS timeout delete mechanism is invoked at user-specified intervals to remove any shipped terminal definitions that have not been used for longer than a user-specified time.

You can use the **SET DELETSHIPPED** command to change the values that control the mechanism.

Press the Clear key to clear the screen. There are two ways to start this transaction:

- Type `CEMT SET DELETSHIPPED` (or suitable abbreviations for the keywords). The resulting display lists the current status, similar to that obtained by using the CEMT INQUIRE DELETSHIPPED command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET DELETSHIPPED` (or suitable abbreviations for the keywords), followed by one or more attributes that you want to change. For example, `cemt s de idl(0150000)` specifies that shipped terminal definitions are to remain installed for at least fifteen hours after they become inactive.

CEMT SET DELETSHIPPED



Options

IDLE(*value*)
Specifies, in the form 0hhmmss+, the minimum time that an inactive shipped terminal definition must remain installed in this region. When the CICS timeout delete mechanism is invoked, only those shipped definitions that have been inactive for longer than this time are deleted.

The time interval can be in the range 00 - 99 hours; 00 - 59 minutes; and 00 - 59 seconds.

At CICS startup, the interval is set to the value specified on the DSHIPIDL system initialization parameter.

INTERVAL(*value*)
Specifies, in the form 0hhmmss+, the interval between invocations of the CICS timeout delete mechanism. The timeout delete mechanism removes any shipped terminal definitions that have not been used for longer than the time specified by the IDLE option.

The time interval can be in the range 00 - 99 hours; 00 - 59 minutes; and 00 - 59 seconds. If you specify 0, the timeout delete mechanism is not invoked.

At CICS startup, the interval is set to the value specified on the DSHIPINT system initialization parameter. By resetting the interval, you can change the time of day at which a mass delete operation takes place.

Note: The revised interval starts from the time that the command is issued, not from the time that the timeout delete mechanism was last invoked, nor from the time of CICS startup.

CEMT SET DISPATCHER

Change the system values used by CICS dispatcher.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

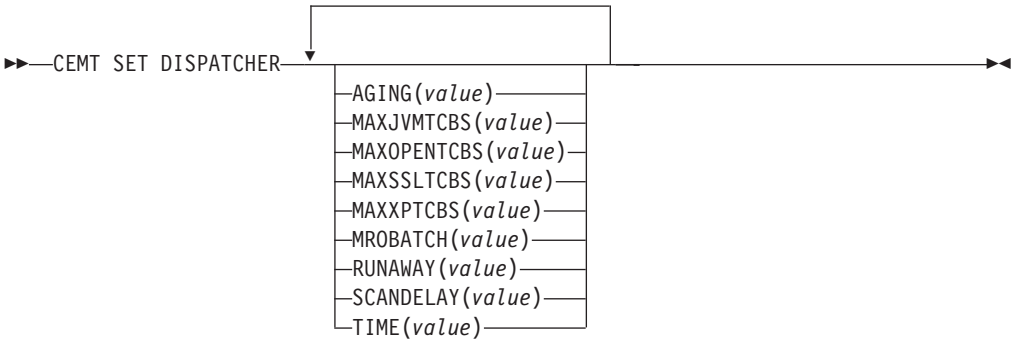
Syntax

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET DISPATCHER` (or suitable abbreviations ofr the keywords). The resulting display lists the current status, similar to that obtained by using the **`CEMT INQUIRE DISPATCHER`** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET DISPATCHER` (or suitable abbreviations ofr the keywords), followed by one or more attributes that you want to change. For example, `cemt s disp maxo(value) maxt(value)` resets the values for maximum number of L8 and L9 mode TCBs and the maximum number of tasks.

Typing `?` at the beginning of either the first or second line gives a syntax prompt.

CEMT SET DISPATCHER



Options

AGING (*value*)
Specifies the time factor for CICS to use in the priority aging algorithm to increment the priority of a task. The value is expressed as "milliseconds per unit of priority", and must be in the range 00000 - 65535. The priority aging factor is defined initially by the PRTYAGE system initialization parameter, in which the value represents the number of milliseconds that must elapse before the priority of a waiting task can be adjusted upward by 1. See the *CICS System Definition Guide* for information about the PRTYAGE system initialization parameter.

MAXJVMTCBS(*value*)
Specifies the maximum number of J8 and J9 mode open TCBs that can exist concurrently in the CICS region. The value specified can be in the range 1 - 999. If you reduce MAXJVMTCBS from its previously defined value, and the new value is less than the number of open TCBs currently allocated, CICS detaches TCBs to achieve the new limit only when they are freed by user tasks. Transactions are not abended to allow TCBs to be detached to achieve the new limit. If there are tasks queued waiting for a J8 mode TCB and you increase MAXJVMTCBS from its previously defined value, CICS attaches a new TCB to resume each queued task, up to the new limit.

MAXOPENTCBS(*value*)
Specifies the maximum number of L8 and L9 mode open TCBs that can exist concurrently in the CICS region. The value specified can be in the range 1 - 2000. If you reduce MAXOPENTCBS from its previously defined value, and the new value is less than the number of open TCBs currently allocated, CICS detaches TCBs to achieve the new limit only when they are freed by user tasks. Transactions are not abended to allow TCBs to be detached to achieve the new limit. If there are tasks queued waiting for an L8 or L9 mode TCB and you increase MAXOPENTCBS from its previously defined value, CICS attaches a new TCB to resume each queued task, up to the new limit.

MAXSSLTCBS(*value*)
Specifies the maximum number of S8 mode open TCBs that can exist concurrently in the CICS region. The value specified can be in the range 1 - 1024.

MAXXPTCBS(*value*)
Specifies the maximum number of X8 and X9 mode open TCBs that can exist concurrently in the CICS region. The value specified can be in the range 1 - 999. If you reduce MAXXPTCBS from its previously defined value, and the

new value is less than the number of open TCBs currently allocated, CICS detaches TCBs to achieve the new limit only when they are freed by user tasks. Transactions are not abended to allow TCBs to be detached to achieve the new limit. If there are tasks queued waiting for an X8 or X9 mode TCB and you increase MAXXPTCBS from its previously defined value, CICS attaches a new TCB to resume each queued task, up to the new limit.

MROBATCH (*value***)**
Specifies the number of MRO requests from connected regions that are to be batched before this region is posted. The value is in the range 001 - 255.

RUNAWAY (*value***)**
Specifies the interval, in milliseconds, for which a task can have control before it is assumed to be looping.

You can specify 0, or a value in the range 500 - 2700000. The value chosen is rounded down to a multiple of 500. CICS purges a task if it has not given up control after this interval (that is, if the task appears to be looping). If you specify zero, runaway task control is inoperative (that is, tasks do not get purged if they appear to be looping). The value is independent of, and can be less than, the value in the TIME operand.

CICS runaway-task detection is based upon task time, that is, the interval is decremented only when the task has control of the processor. You do not, therefore, need to allow for the effect of other jobs when setting the value.

SCANDELAY (*value***)**
Specifies the maximum number of milliseconds between the receipt of a z/OS Communications Server request, and the time when CICS begins to process the request. The number must be in the range 0 - 5000, and must be less than or equal to the TIME value.

In the summary report of the dispatcher statistics, SCANDELAY is referred to as ICVTSD.

TIME (*value***)**
Specifies the interval, in milliseconds, for which CICS releases control to the operating system if no transactions are ready to resume processing. This interval is known as the *region exit interval*.

It can have a value in the range 100 - 3600000, and must be greater than or equal to the SCANDELAY value.

In the summary report of the dispatcher statistics, TIME is referred to as ICV.

CEMT SET DOCTEMPLATE

Refresh the cached copy of a document template installed in your CICS region, or phase in a new copy of a CICS program or exit program that is defined as a document template.

In the CICS Explorer, the Document templates operations view provides a functional equivalent to this command.

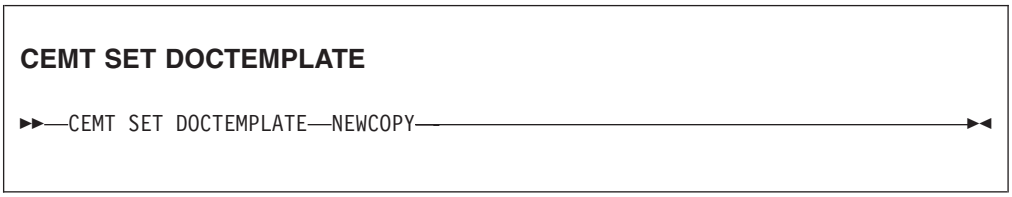
Syntax

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET DOCTEMPLATE` followed by one or more DOCTEMPLATE resource definition names, or ALL. The resulting display lists the current status, similar to that obtained by using the **CEMT INQUIRE DOCTEMPLATE** command.

- Type CEMT SET DOCTEMPLATE, followed by one or more DOCTEMPLATE resource definition names or ALL, then followed by NEWCOPY.

Typing ? at the beginning of either the first or second line gives a syntax prompt.



Options

ALL
The action you request is performed for all resources of the specified type that you are authorized to access.

(*value***)**
Specifies the 1- to 8-character name of the DOCTEMPLATE resource definition.

NEWCOPY
Specifies that if a cached copy of the document template exists, it is to be deleted. If the document template resides in a CICS program or exit program, a new copy of the program is to be phased in. If caching is required for the document template, a new copy of the document template is to be loaded into the cache.

CEMT SET DSAS

Change the system storage attributes for dynamic storage areas (DSAs) below 16 MB and DSAs above 16 MB but below 2 GB.

In the CICS Explorer, the Global Dynamic Storage Areas operations view provides a functional equivalent to this command.

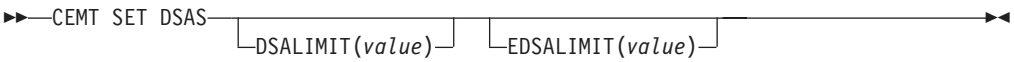
Syntax

Press the Clear key to clear the screen. There are two ways to start this transaction:

- Type CEMT SET DSAS (or suitable abbreviations for the keywords). The resulting display shows the current status, similar to that obtained by using the **CEMT INQUIRE DSAS** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type CEMT SET DSAS (or suitable abbreviations for the keywords), followed by one or more attribute settings that you want to change. For example, cemt s dsa ds(*value*) ed(*value*) resets the value for dynamic storage below 16 MB (below the line) and the value for dynamic storage above 16 MB but below 2 GB (above the line).

Typing ? at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET DSAS



Options

DSALIMIT(*value***)**
Specifies the maximum amount of storage, as a total number of bytes, within which CICS can dynamically allocate storage for the four individual DSAs that reside below 16 MB (below the line). For more information, see DSALIM system initialization parameter in the System Definition Guide. If DSALIMIT specifies a value lower than the current limit, CICS might not implement the new limit immediately, but attempts to do so over time as dynamic storage is freed in the individual DSAs.

EDSALIMIT(*value***)**
Specifies the maximum amount of storage, as a total number of bytes, within which CICS can dynamically allocate storage for the individual DSAs that reside above 16 MB but below 2 GB (above the line). For more information, see EDSALIM system initialization parameter in the System Definition Guide. If EDSALIMIT specifies a value lower than the current limit, CICS might not implement the new limit immediately, but attempts to do so over time as dynamic storage is freed in the individual DSAs.

CEMT SET DSNAME

Change the status of an external data set.

Description

You can use the **SET DSNAME** command to change the status of a VSAM base data set. You can use the command to perform the following actions on a data set:

- Mark a VSAM data set as quiesced, or unquiesced, throughout the sysplex.
- Make a VSAM data set available or unavailable to a CICS region. (The availability function does not operate across the sysplex—a SET DSNAME(...) AVAILABILITY command is effective only within the CICS region in which it is issued.)
- Retry all unit of work (UOW) log records that are shunted because of the failures of this data set (other than indoubt failures).
- Force any UOWs that are shunted because of indoubt failures, and which have updated this data set, to complete.
- Purge shunted UOW log records that hold retained locks (other than those because of indoubt failures) for any records in the data set, and release the retained locks.
- Cancel any attempt to recover lost RLS locks for the data set using the UOWACTION and ACTION(RESETLOCKS) options.

The SET DSNAME command does not distinguish between data sets accessed in RLS and non-RLS mode.

The new options for the SET DSNAME command are subject to the following rules relating to the order of processing and the combinations of keywords and multiple keywords on the same command:

- If REMOVE is specified, no other attribute is allowed.

- Options are processed in the following order:
  1. RECOVERED
  2. UNQUIESCED
  3. AVAILABLE
  4. RETRY
  5. UOWACTION
  6. RESETLOCKS
  7. UNAVAILABLE
  8. QUIESCED

If you specify RETRY, do not also specify UNAVAILABLE or QUIESCED, because this would cause backout retries to fail. UNQUIESCE should not be specified with other attributes, because the other options might fail.

Some attributes of a data set cannot be set until the first file that references the data set is opened. Where an attribute is not valid until a file is opened, an error is returned. QUIESCESTATE is an attribute that can be set before any files are opened against the specified data set.

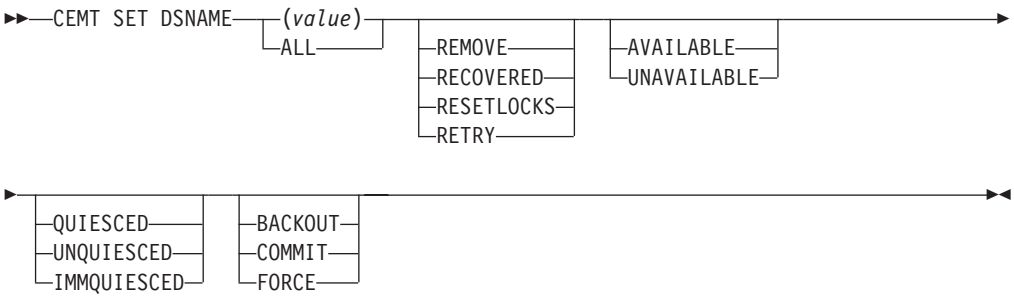
Syntax

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET DSNAME` (or suitable abbreviations for the keywords), followed by either a value that corresponds to a data set name, or ALL. The resulting display lists the current status, similar to that obtained by using the **CEMT INQUIRE DSNAME** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET DSNAME` (or suitable abbreviations for the keywords), followed by a value that corresponds to a data set name or ALL, then followed by one or more attributes that you want to change. For example, `cemt s dsn(applb.*) q` quiesces all data sets with a high-level qualifier of `APPLB` in all CICS regions within the sysplex.

Resetting the values takes effect immediately.

CEMT SET DSNAME



Options

(*value***)**
Specifies one or more names (1 - 44 characters) of an external data set.

ALL
Any change you request is made to all resources of the specified type that you are authorized to access.

AVAILABLE
Specifies whether the data set is marked in this CICS region as available for use. This command sets the availability indicator, which is a local flag that a CICS region maintains in a data set name block (DSNB) for each data set. CICS can issue either RLS or non-RLS open requests for this data set.

BACKOUT
Specifies that all shunted indoubt UOWs that hold locks on this data set are backed out.

COMMIT
Specifies that all shunted indoubt UOWs that hold locks on this data set are committed.

FORCE
Specifies that all shunted indoubt UOWs that hold locks on this data set are forced to back out or commit, as specified by the ACTION attribute on the transaction resource definition.

IMMQUIESCED
Causes all existing CICS open RLS-mode files throughout the sysplex to close and the data set to be marked as quiesced in the ICF catalog. Each CICS in the sysplex abends all in-flight UOWs that are accessing the data set before closing files, causing in-flight UOWs to back out. Any UOWs that fail backout are shunted. No RLS-mode files can open against this data set, but non-RLS-mode open requests are permitted.

IMMQUIESCED also sets the file state to UNENABLED if it was ENABLED. A subsequent **SET DSNAME UNQUIESCED** command restores the file state to ENABLED, if it was set UNENABLED by a QUIESCED or IMMQUIESCED action, but *not* if the UNENABLED state is because of another event. This state change is recorded in the CICS global catalog.

Note: Using the IMMQUIESCED option causes any tasks currently using the data set to be terminated immediately, using the CICS task FORCEPURGE mechanism. In some extreme cases CICS might terminate abnormally. For this reason, setting a data set as quiesced using the IMMQUIESCED option should be restricted to exceptional circumstances.

QUIESCED
Causes all existing CICS files open in RLS mode throughout the sysplex to close and the data set to be marked as quiesced in the ICF catalog. Each CICS in the sysplex waits until all in-flight UOWs that are accessing the data set have reached syncpoint before closing the files. That is, the UOWs are in one of the following states:

- Successfully committed
- Successfully backed out
- Shunted because they failed indoubt
- Shunted because they failed commit
- Shunted because they failed backout

No files can open in RLS mode against this data set, but non-RLS open requests are permitted.

QUIESCED also sets the file state to UNENABLED if it was ENABLED. A subsequent **SET DSNAME UNQUIESCED** command restores the file state to ENABLED, if it was set UNENABLED by a QUIESCED or IMMQUIESCED action, but *not* if the UNENABLED state is because of another event. This state change is recorded in the CICS global catalog.

RECOVERED
The data set has been restored from a backup version and forward recovery has run and completed successfully. CICS normally updates the backup-while-open (BWO) attributes for the base cluster data set BASEDSNAME, using MVS/DFP 3.2 Callable Services.

If the BWO attributes are not updated after the restore of a backup copy of a data set that was taken using BWO, CICS fails a subsequent file open because the data set is marked as down-level.

REMOVE
This option is used when a data set is no longer required on the local system. It deletes the data set name block for the named data set. The REMOVE option can be used only when there are no file control resource definition entries that refer to the specified data set. If any file definitions refer to the data set name when you enter **SET DSNAME REMOVE**, you receive an error message indicating that the use count of the data set name block is not zero. To ensure that no file definitions refer to the data set name:

- Close all files that refer to the data set. These files can be found using the command:

```
CEMT INQUIRE FILE(*) DSNAME(dsname)
```

- Either remove the data set name from each file entry (by over-typing the data set name with blanks on the CEMT display), or discard the file definition entirely.

If you have a base data set with one or more alternate indexes, you cannot remove the data set name block for the base until you have removed all references to the base data set and all references to paths.

RESETLOCKS (*VSAM only***)**
Purges shunted UOW log records for backout-failed and commit-failed UOWs that hold locks on this data set, and releases the retained locks:

- Backout-failed UOWs are those that failed during backout processing.
- Commit-failed UOWs are those that have updated RLS data sets, and have failed to release locks during the second phase of 2-phase commit syncpoint processing.

If you specify this option, you are accepting backout failure and some loss of data integrity rather than retaining locks and delaying transactions, and therefore use it only as a last resort.

For backout-failed and commit-failed UOWs that hold locks on the data set, all records relating to this data set are removed from the system log and all retained record locks held by this CICS for the data set are released. Diagnostic messages are written to the CSFL transient data queue for each backout-failed log record that is removed as a result of the RESETLOCKS operation.

RESETLOCKS applies only to shunted UOW log records that are classed as backout-failed and commit-failed. Therefore it cannot be issued for a BDAM data set. Backout-failed and commit-failed log records are not shunted for BDAM data sets.

You might use RESETLOCKS if backout-failed or commit-failed log records are holding up lost locks recovery for the data set, and there is no other way of resolving them.

Note:

1. This option does not apply to shunted *indoubt* UOWs. Try to resolve the shunted indoubt UOWs that hold locks on the data set in other ways before issuing RESETLOCKS; for example, by using COMMIT, BACKOUT, or FORCEUOW.
2. RESETLOCKS can fail during the commit phase (for example, if an error occurs while CICS is trying to release the RLS locks), in which case the UOWs revert to being shunted as commit-failed UOWs.

RETRY
Specifies that shunted UOW log records, caused by failed backout and commit processing as a result of the failure of this data set, are retried. This is similar in operation to the **EXEC CICS RESYNC** command, but applies to backout-failed and commit-failed UOWs only, and not to indoubt UOWs.

Use RETRY when the data set has shunted backout- or commit-failed UOWs associated with it, and you believe that some or all of the data set problems are either transient or have been resolved.

Messages that are issued at the time of a data set failure that causes UOWs to be shunted recommend the actions that are required to recover from the failure.

RETRY does not affect data integrity, and can be used safely at any time to enable some failed recovery work to complete.

UNAVAILABLE
Marks the data set as unavailable, and prevents the data set being opened in either RLS or non-RLS modes. This command sets or unsets the availability indicator, which is a local flag that a CICS region maintains in a data set name block (DSNB) for each data set.

UNQUIESCED
Causes the data set to be marked as unquiesced in the ICF catalog. RLS or non-RLS mode files can be opened against this data set, the access mode (RLS or non-RLS) being established by the first open.

If a file has been set UNENABLED by a SET DSNAME IMMQUIESCED or QUIESCED command, UNQUIESCED sets it back to ENABLED, provided that there have been no other explicit changes to the enablement state in the meantime. This state change is recorded in the CICS global catalog.

CEMT SET DUMPDS

Change the attributes of a CICS dump data set.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

You can use the **SET DUMPDS** command to open or close the active CICS dump data set, and specify whether the automatic switch to the inactive dump data set occurs when the active dump data set is full.

For more information about dumps, see the *CICS Problem Determination Guide*.

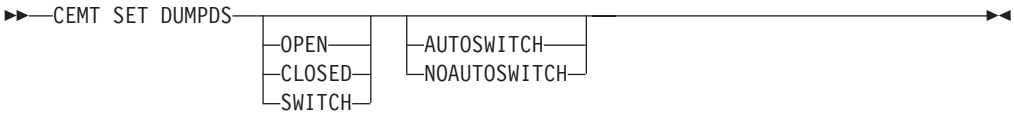
Syntax

Press the Clear key to clear the screen. There are two ways to start this transaction:

- Type `CEMT SET DUMPDS` (or suitable abbreviations for the keywords). The resulting display lists the current status, similar to that obtained by using the **CEMT INQUIRE DUMPDS** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET DUMPDS` (or suitable abbreviations for the keywords), followed by one or more attribute settings that you want to change. For example, `cemt s du o a` opens the active dump data set and switches to the inactive dump data set only when the active dump data set is full.

Typing `?` at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET DUMPDS



Options

AUTOSWITCH
The switch to the inactive dump data set occurs only when the active dump data set is full, but not subsequently.

CLOSED
The active CICS dump data set is closed.

NOAUTOSWITCH
When the active dump data set is full, there is not to be an automatic switch to the inactive dump data set.

OPEN
The active CICS dump data set is opened.

SWITCH
The currently active dump data set is to become inactive, and the currently inactive data set is to become active.

CEMT SET ENQMODEL

Set the status of enqmodel definitions on the local system.

Description

Each ENQMODEL definition may be in one of the following states

ENABLED
Matching ENQ requests will be processed in the normal way.

DISABLED
Matching ENQ requests will be rejected, and the issuing task abended abcode ENQ_DISABLED. Matching INSTALL/CREATE/DISCARD requests will be processed.

WAITING
Matching ENQ requests will be rejected, and the issuing task abended abcode ENQ_DISABLED. There are INSTALL/CREATE/DISCARD requests waiting to be processed.

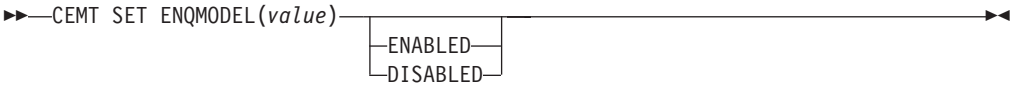
This command provides a means for you to set the ENABLED and DISABLED states for ENQmodel resources installed on the local system. An enqmodel must be enabled to allow matching EXEC ENQ requests to be processed. It must be disabled to allow a more specific enqmodel to be enabled.

Enqmodels forming nested generic enqnames must be enabled in order, from the most to the least specific. For example, enable ABCD* then ABC* then AB*. If you attempt to enable a more specific enqmodel when a less specific enqmodel is already enabled, the result is that message DFHNQ0107 is issued.

You cannot enable/disable an enqmodel which is in the waiting state.

Syntax

CEMT SET ENQMODEL



Options

(*value*)
The 8-character identifier of the resource definition

DISABLED
The ENQMODEL will be put into the WAITING state (See Inquire Enqmodel) until there are no enqueues in the local system which match the ENQNAME pattern. It will then be DISABLED. Once DISabled, matching ENQ requests will be rejected, and the issuing task abended abcode ENQ_DISABLED.

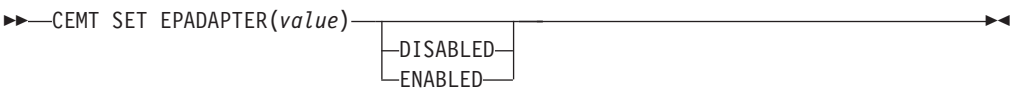
ENABLED
If the enqmodel is DISABLED, it will be ENABLED. Once enabled, matching ENQ requests will be processed in the normal way.

CEMT SET EPADAPTER

Enable or disable an EP adapter.

In the CICS Explorer, the EP adapter operations view provides a functional equivalent to this command.

CEMT SET EPADAPTER



Options

(*value*)
The 32-character name of an EP adapter.

DISABLED
Specifies that the EP adapter is to be disabled.

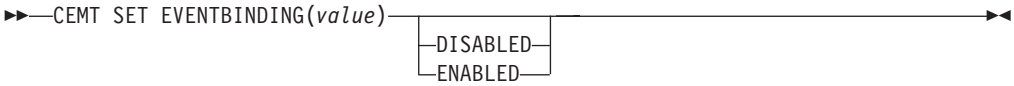
ENABLED
Specifies that the EP adapter is to be enabled.

CEMT SET EVENTBINDING

Enable or disable an event binding.

In the CICS Explorer, the Events bindings operations view provides a functional equivalent to this command.

CEMT SET EVENTBINDING



Options

(*value*)
The 32-character name of an event binding.

DISABLED
specifies that the event binding is to be disabled.

ENABLED
specifies that the event binding is to be enabled. You can enable an event binding that is in any state except DISABLING or DISCARDING.

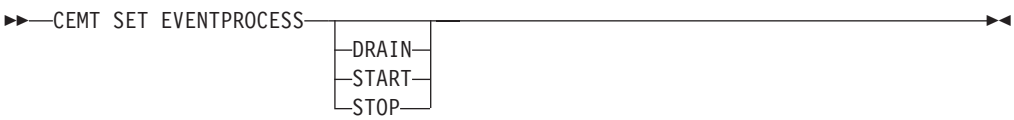
CEMT SET EVENTPROCESS

Change the status of event processing.

Note: Do not change the status of event processing (that is, set to start, drain, or stop) while a unit of work that captures synchronous transactional events is in progress because you might cause the events to be backed out and the transaction to end abnormally.

In the CICS Explorer, the Event processing operations view provides a functional equivalent to this command.

CEMT SET EVENTPROCESS



Options

DRAIN
Set the status of event processing to draining. Event capture is stopped immediately.

Any transactional events on the dispatcher queue will be deleted. Transactional events are not considered captured until a syncpoint occurs and a syncpoint event will not now be captured.

Any non-transactional events will be emitted. When the last event on the queue is emitted, the state of event processing changes to stopped.

START
Set the status of event processing to started. For in-flight transactions, the capture of non-transactional events starts immediately and the capture of transactional events starts at the next sync point.

STOP
Set the status of event processing to stopped. Event capture is stopped immediately.

All events on the dispatcher queue are deleted.

CEMT SET FEPI

Comments

For information on **CEMT SET FECONNECTION**, **CEMT SET FEPOOL**, **CEMT SET FEPROPSET**, and **CEMT SET FETARGET**, see the *CICS Front End Programming Interface User's Guide*.

CEMT SET FILE

Change some attributes of one or more VSAM or BDAM files.

In the CICS Explorer, the Files operations view provides a functional equivalent to this command.

Syntax

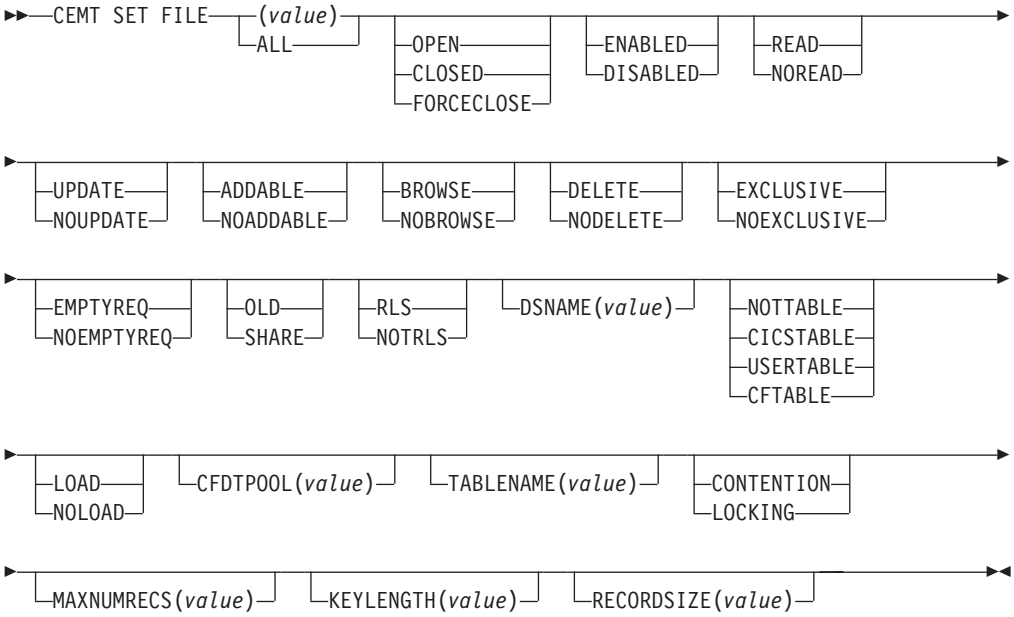
Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET FILE` (or suitable abbreviations for the keywords), followed by a value that corresponds to a file name, or ALL. The resulting display lists the current status, similar to that obtained by **CEMT INQUIRE FILE**. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET FILE` (or suitable abbreviations for the keywords), followed by a value that corresponds to a file name or ALL, then followed by one or more attribute settings that you want to change. For example, `cemt s fi al op en` attempts to set all files open and enabled for use.

Typing `?` at the beginning of either the first or second line gives a syntax prompt. Reset values take effect immediately.

Note: Some attributes can be changed only when the specified file is closed and disabled. In these cases, the reset values take effect when the file is enabled for use again.

CEMT SET FILE



Options

(*value***)**
One or more file names (1 - 8 characters).

ADDABLE
You can add records to the file.

ALL
Any change you request is made to all files that you are authorized to access.

BROWSE
You can browse records in the file.

CFDTPOOL (*value***)**
Specifies the name of the CFDT pool in which the coupling facility data table resides.

CFTABLE
The file name represents a coupling facility data table.

CICSTABLE
The file name represents a CICS-maintained data table.

CLOSED
The file is closed; the access method does not allow accesses to the data until the file is opened. The file can be opened either explicitly by the user or, if the file has the ENABLED attribute, implicitly by CICS on the next reference to the file.

When the SET FILE CLOSED command is used, the file is closed and also disabled (set UNENABLED). (A file disabled in this way prevents access requests from implicitly opening the file again.) The close is effected at the time of the command only if there are no tasks currently accessing the file. If there are current users, the display indicates CLOSE requested. This means that, when the last user finishes using the file, it effects the close.

A file that has been disabled by a SET FILE CLOSED command becomes enabled again by a subsequent SET FILE OPEN command. To distinguish a file that has been disabled by a SET FILE CLOSED from one that has been disabled by a SET FILE DISABLED command, the SET FILE CLOSED command closes the file with the UNENABLED attribute.

A file with the DISABLED attribute can be enabled only by a SET FILE ENABLED command. A file with the UNENABLED attribute can be enabled by a SET FILE ENABLED or SET FILE OPEN command.

CONTENTION (CFDT ONLY)
The coupling facility data table is to use the contention model, in which records are not locked when they are read for update, but an error is returned on a subsequent REWRITE or DELETE if the record has changed or been deleted since it was read for update.

DELETE
You can delete records from the file.

DISABLED
The file is not available for use by transactions except for those that are currently using it. If there are any such users, "BEING DISABLED" is also displayed. The **CEMT SET FILE DISABLED** command has no effect on existing users; it prevents new users from accessing the file.

The file can be reenabled by a **SET FILE ENABLED** command. (See also UNENABLED.)

It is not possible to disable a remote file.

DSNAME (*value*)
Specifies the name (1 - 44 characters) of the VSAM or BDAM data set with which this file is associated. CEMT SET FILE(*value*) DSNAME(*value*) associates the data set with the file and causes the FILECOUNT of the data set to be incremented by 1. You can use **CEMT SET FILE DSNAME** with a null DSNAME value to dissociate a file from its current data set.

Note: DSNAME replaced OBJECTNAME.

EMPTYREQ (*VSAM only*)
Applies only to a VSAM file that has been defined to VSAM as a cluster with the REUSE parameter, and which is opened in non-RLS access mode. It indicates that when a file is next opened, its data is erased.

If the file is opened in RLS access mode this parameter has no effect.

A **CEMT SET FILE EMPTYREC** command can be applied only to files that are closed and either disabled or unenabled, and has no effect on the data until the file is opened. A **SET FILE EMPTYREQ** command can be canceled by a **SET FILE NOEMPTYREQ** command.

ENABLED
The file is available for use by transactions and, if closed, it is opened on the first request. If the command CEMT SET FILE(*value*) CLOSED ENABLED is given while the file is in use, the status of the file becomes closed-unenabled. If the command CEMT SET FILE(*value*) CLOSED is given while the file is in use, the file is disabled to prevent new users accessing the file.

EXCLUSIVE
Records on the file are protected from simultaneous update by being placed under exclusive control when a read for update is issued.

FORCECLOSE
Similar to SET FILE CLOSED, except that any existing users of the file are abended (abend code AKC3 or ATCH). As for CLOSED, the close is effected by the last of these users.

Note: Closing a file using the FORCECLOSE option causes tasks of any current users of the file to be terminated immediately by the CICS task FORCEPURGE mechanism. Data integrity is not guaranteed with this mechanism. In some extreme cases (for example, if an error occurs during backout processing) CICS might terminate abnormally. For this reason, closing files using the FORCECLOSE option should be restricted to exceptional circumstances.

KEYLENGTH (*value*)
Specifies the key length of records in a coupling facility data table that does not require pre-loading from a source data set. To set a key length, specify a value in the range 1 through 16. To clear a key length (set it to null values), specify KEYLENGTH(0).

You can specify the key length for a file that does not currently refer to a coupling facility data table, but which could be switched to use a coupling facility data table at a later date.

LOAD
The coupling facility data table requires loading from a source data set before it is fully usable; the transactions that use this coupling facility data table rely on it containing the records from the specified source data set.

LOCKING (CFDT ONLY)
The coupling facility data table is to use the locking model, in which records are locked when they are updated.

MAXNUMRECS (*value*)
Specifies the maximum number of records that the data table for this file can hold.

For any type of table, if you want to set a limit, specify a value in the range 1 through 99999999. If you do not want any limit to apply, specify MAXNUMRECS(0), which CICS interprets as no limit, and sets internally to the maximum positive fullword value (+2147483647 or X'7FFFFFFF').

Note: The new value takes effect only when a new data table is created. This happens on each file open for a CICS-maintained or user-maintained data table, but only when the table does not already exist for a CF data table. Once a coupling facility data table has been created, you can change the maximum number of records only through the CFDT server MVS command interface.

To specify MAXNUMRECS for a recoverable coupling facility data table, use a value that is between 5% and 10% more than the maximum number of records that the table is expected to contain. This allows for additional records that might be created internally for processing recoverable requests. The margin to be left for this internal processing depends on the level of use of the coupling facility data table, and the nature of that use. An effect of this is that the NOSPACE condition (with a RESP2 of 102) can be raised on a WRITE or REWRITE request to a recoverable coupling facility data table that apparently has fewer records than the MAXNUMRECS limit specifies.

NOADDABLE
You cannot add records to the file.

NOBROWSE
You cannot browse records in the file.

NODELETE
You cannot delete records from the file.

NOEMPTYREQ (*VSAM only***)**
Applies only to VSAM files. Specifies that when a file is next opened, its data is not erased.

A **CEMT SET FILE NOEMPTYREC** command can be applied only to files that are closed and either disabled or unenabled, and cancels the implicit request in a previous **SET FILE EMPTYREQ** command.

NOEXCLUSIVE
Records on the file are not protected from simultaneous update.

NOLOAD
The coupling facility data table does not require loading from a source data set. It is fully usable as soon as it is created, and is populated entirely by the transactions that use it.

NOREAD
You cannot read records in the file.

NOTRLS (*VSAM only***)**
The file is defined to be accessed in a non-RLS mode (LSR or NSR).

For information about switching from RLS to non-RLS mode, see the Recovery and restart overview in the Recovery and Restart Guide.

NOTTABLE
The file name does not represent a data table.

NOUPDATE
You cannot update records in the file.

OLD
The disposition of the data set associated with the file is defined as unshared for dynamic allocation.

OPEN
The file is open. If the file is OPEN ENABLED, it is available for data accesses by CICS transactions. If it is OPEN DISABLED, it must first be enabled before it is available.

If the file was unenabled by a previous SET FILE CLOSED command, the **SET FILE OPEN** command reenables the file implicitly.

READ
You can read records in the file.

RECORDSIZE
Specifies the maximum record size for a coupling facility data table that does not require pre-loading from a source data set.

You can specify the record size for a file that does not currently refer to a coupling facility data table, but which could be switched to use a coupling facility data table at a later date. Specify a record size of 1 to 32767 bytes, or specify a record size of zero to remove a previously defined value.

RLS (*VSAM only***)**
The file is defined to be accessed in RLS mode.

SHARE
The disposition of the data set associated with the file is defined as shared for dynamic allocation.

TABLENAME (*value***)**
Specifies the name of the table for a coupling facility data table.

UPDATE
You can update records in the file.

USERTABLE
The file name represents a user-maintained data table.

CEMT SET GTFTRACE

Change the status of GTF tracing.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

You can use the **SET GTFTRACE** command to switch CICS tracing to the MVS Generalized Tracing Facility (GTF) on and off. This command is intended for use at terminals that do not support the CETR full-screen tracing transaction.

For guidance on using GTF tracing, see the *CICS Problem Determination Guide*.

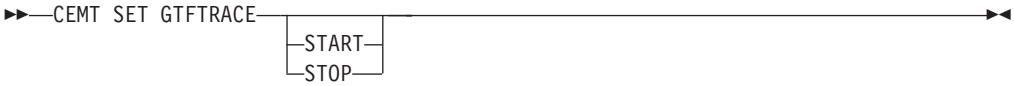
Syntax

Press the Clear key to clear the screen. There are two ways to start this transaction:

- Type `CEMT SET GTFTRACE` (or suitable abbreviations for the keywords). The resulting display lists the current status, similar to that obtained by using the **CEMT INQUIRE GTFTRACE** command. You can tab to the highlighted field and overtype it with the required value.
- Type `CEMT SET GTFTRACE` (or suitable abbreviations for the keywords), followed by the attribute setting that you want to change. For example, `cemt s g sta` starts CICS GTF tracing.

Typing `?` at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET GTFTRACE



Options

START
CICS tracing to the MVS Generalized Tracing Facility (GTF) is started.

STOP
CICS tracing to the MVS Generalized Tracing Facility (GTF) is stopped.

CEMT SET HOST

Enable or disable a virtual host.

Description

The SET HOST command is used to set the status of a virtual host to enabled or disabled. Disabling a virtual host means that all the URIMAP definitions that make up the virtual host cannot be accessed. When a virtual host is disabled, CICS returns an HTTP response with a 503 (Service Unavailable) status code to Web clients.

When you inquire on an individual URIMAP definition, a special status HDISABLED is returned to indicate that the virtual host is disabled. You do not need to change the disabled status of the URIMAP definitions individually; the SET HOST command can be used to re-enable all the URIMAP definitions that make up the virtual host. However, note that a URIMAP definition with the HDISABLED status cannot be discarded. If you want to discard the definition, it must be disabled individually (using the SET URIMAP command).

Syntax

CEMT SET HOST

Options

(*value***)**
specifies the name of a virtual host. The name of each virtual host is taken from the host name specified in the URIMAP definitions that make up the virtual host. For example, if your CICS region contained URIMAP definitions that specified a host name of `www.example.com`, CICS would create a virtual host with the same name. A host name in a URIMAP definition can be up to 120 characters.

ENABLED
The URIMAP definitions that make up the virtual host can be accessed.

DISABLED
The URIMAP definitions that make up the virtual host cannot be accessed.

TCPIPSERVICE((*value***)**
The TCPIPSERVICE resource that specifies the inbound port to which this virtual host relates. If you do not specify a value, the virtual host relates to all TCPIPSERVICE resources.

CEMT SET INTTRACE

Switch CICS internal tracing on or off.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

You can use SET INTTRACE to switch CICS internal tracing on and off. This command is intended for use at terminals that do not support the full-screen tracing transaction CETR.

Syntax

Press the Clear key to clear the screen. There are two ways to start this transaction:

- Type `CEMT SET INTTRACE` (or suitable abbreviations for the keywords). The resulting display lists the current status, similar to that obtained by using the CEMT INQUIRE INTTRACE command. You can tab to the highlighted field and overtype it with the required value.
- Type `CEMT SET INTTRACE` (or suitable abbreviations for the keywords), followed by the attribute setting that you want to change. For example, `cemt s in sta` switches CICS internal tracing on.

Typing `?` at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET INTTRACE

Options

START
CICS internal tracing is started.

STOP
CICS internal tracing is stopped.

CEMT SET IPCONN

Change the attributes of an IPCONN resource or cancel outstanding automatic initiate descriptors (AIDs).

In the CICS Explorer, the IPIC connections operations view provides a functional equivalent to this command.

Description

You can use the **SET IPCONN** command to change some attributes of an IPCONN resource.

To install a new IPCONN definition when one is already installed, you must set the connection to OUTSERVICE before you use the CEDA INSTALL commands for your new definition.

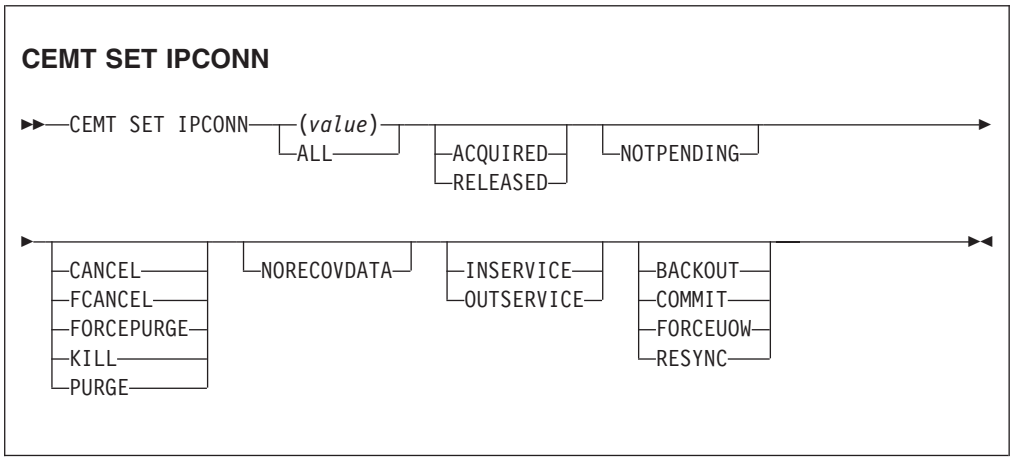
For information about the different kinds of intercommunication connections, see the *CICS Intercommunication Guide*.

Syntax

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET IPCONN` (or suitable abbreviations for the keywords), followed by one or more connection identifiers, or ALL. The resulting display lists the current status, similar to that obtained by using the **CEMT INQUIRE IPCONN** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET IPCONN` (or suitable abbreviations for the keywords), followed by one or more connection identifiers or ALL, then followed by one or more attribute settings that you want to change. For example, `cemt s ip al i` resets the values for all connections to make them available for use (inservice).

Typing `?` at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.



Options

(*value***)**
Specifies the name (1 - 8 characters) of the connection to modify. This is the name of the remote system or region specified on the IPCONN option of the IPCONN definition. You can specify more than one name.

ACQUIRED
Specifies that CICS acquires a session with the system represented by the IPCONN name. A connection cannot be both ACQUIRED and OUTSERVICE.

ALL
Specifies that any changes you request are made to all resources of the specified type that you are authorized to access.

BACKOUT
Specifies that all units of work (UOWs) shunted because of the failure of this connection are backed out. The normal resynchronization process is partially overridden: decisions are taken for any UOWs that are indoubt because of a failure of the connection, but the decisions are recorded and any data inconsistencies are reported when the connection is next acquired.

CANCEL

Specifies that queued requests by transactions to use this IPCONN are canceled.

Queued requests to use this IPCONN by CICS system transactions that manage communications across the IPCONN are not purged unless FCANCEL is specified.

Message DFHISnnnn is written to CSMT to indicate how many queued requests to use this IPCONN have been deleted for the IPCONN and how many remain.

A "QUEUED REQUESTS CANCELED" message appears on the CEMT panel whenever queued requests to use this IPCONN are deleted using the CANCEL option of the **CEMT SET IPCONN** command.

COMMIT
Specifies that all UOWs shunted because of the failure of this connection are committed. The normal resynchronization process is partially overridden: decisions are taken for any units of work that are indoubt because of a failure of the connection, but the decisions are recorded and any data inconsistencies are reported when the connection is next acquired.

FCANCEL
Specifies that all queued requests by transactions to use this IPCONN are canceled, including requests by CICS system transactions that manage communications across this IPCONN. This can lead to unpredictable results and should be used only in exceptional circumstances.

A "QUEUED REQUESTS CANCELED" message appears on the CEMT panel whenever queued requests to use this IPCONN are deleted using the FCANCEL option of the CEMT SET IPCONN command.

FORCEPURGE
Specifies that all transactions running on sessions to the connected system are abnormally terminated immediately. This can lead to unpredictable results and should be used only in exceptional circumstances.

In some extreme cases (for example, if an error occurs during backout processing), CICS might terminate abnormally.

FORCEUOW
Specifies that all UOWs shunted because of the failure of this connection are forced to back out or commit, as specified in the ACTION option of the TRANSACTION definition. The normal resynchronization process is partially overridden: decisions are taken for any units of work that are indoubt because of a failure of the connection, but the decisions are recorded and any data inconsistencies are reported when the connection is next acquired.

INSERVICE
Specifies that the system is placed in service; that is, available for use.

KILL
Specifies that the task is terminated. System and data integrity is not guaranteed. The KILL option extends the PURGE and FORCEPURGE options. Use this option only after first attempting a purge or forced purge. The KILL option does not guarantee integrity of any kind, but in some situations you can use it to free up a stalled region, thus enabling the region to continue processing. In some cases, for example, if a task is killed during backout processing, CICS terminates abnormally.

NORECOVDATA
Specifies that the normal resynchronization process is overridden. NORECOVDATA forces indoubt units of work (according to the transaction definitions), targets any resynchronization tasks that are outstanding for the

connection, and erases the logname previously received from the partner system. The state of the connection is reset.

Note: Use SET IPCONN NORECOVDATA only in exceptional circumstances. this option erases recovery information and might compromise data integrity for units of work that have updated resources on remote systems.

Examples of circumstances in which you might need to use this option are as follows:

- You need to discard a connection and it is not possible for the quiesce protocols with the partner system completed.
- An operational or logic error results in a logname mismatch for the connection. The connection state must be reset to allow the exchange lognames process to complete.

Note: If you specify NORECOVDATA, you cannot specify COMMIT, BACKOUT, FORCEUOW, RESYNC, or NOTPENDING.

NOTPENDING
For a connection to a CICS Transaction Server for z/OS partner that has performed an initial start, specifies that the normal resynchronization process is overridden.

NOTPENDING forces all indoubt units of work (according to the transaction definition) that were created by the connection before the initial start of the partner. It also forgets any resynchronization tasks (waitforget UOW-links) that are outstanding for the connection, and created before the initial start of the partner.

The PENDING condition indicates the existence of recovery information (either shunted UOWs or decisions remembered for the partner) on a connection that has experienced a lognames mismatch with its partner. This indicates that the partner has performed an initial start and that the recovery protocol has been corrupted by a loss of log data at the partner.

It is not possible to set a connection to NOTPENDING state (forcing indoubt and erasing NOFORGET UOWs) until CICS has made contact with the partner and received a new logname from it.

Decisions for a whole connection can be forgotten, but that does not affect the memory of a decision for any other connection involved in the UOW.

Note: If you specify NOTPENDING you cannot specify COMMIT, BACKOUT, FORCEUOW, RESYNC, or NORECOVDATA. For advice on which option to use, see the notes following the description of the RESYNC option.

The exchange lognames function and the resynchronization function are described in the *CICS Intercommunication Guide* and *SNA LU 6.2 Peer Protocols*.

OUTSERVICE
Place the system out of service; that is, unavailable for use.

PURGE
Specifies that transactions running on the connected system are abnormally terminated. Transactions are terminated only if system and data integrity can be maintained. A transaction is not purged if its definition specifies SPURGE=NO.

RELEASED
Specifies that CICS releases a session with the system represented by the IPCONN name.

RESYNC
Specifies that any UOWs shunted because of the failure of this connection are retried. That is, exchange lognames resynchronization for this connection is attempted. This process should normally start automatically when a connection is acquired or when a UOW is unshunted. The normal resynchronization process is partially overridden: decisions are taken for any units of work that are indoubt because of a failure of the connection, but the decisions are recorded and any data inconsistencies are reported when the connection is next acquired.

Note:

1. The COMMIT, BACKOUT, FORCEUOW and RESYNC operations are synchronous with setting the state of the UOW; that is, an INQUIRE UOW following SET IPCONN BACKOUT, COMMIT, FORCEUOW, or RESYNC returns the new UOW states.
2. Specifying one of these options unshunts all units of work that have failed because of a failure of the connection. Before you issue a **SET IPCONN FORCEUOW** command, you can use the **SET UOW** command to specify commit or back out for each indoubt unit of work explicitly, rather than allowing the default action. Local procedures determine the importance of the data and the method of using the **INQUIRE UOW**, **INQUIRE UOWENQ**, and **INQUIRE UOWLINK** commands to establish the correct actions.
3. You can specify only one of the BACKOUT, COMMIT, FORCEUOW, RESYNC, NOTPENDING, and NORECOVDATA options. Use the **SET IPCONN NORECOVDATA** command only in exceptional circumstances.
4. To force all indoubt units of work caused by a failure of the connection in the same direction, use **SET IPCONN COMMIT** or **SET IPCONN BACKOUT**.
5. The BACKOUT, COMMIT, FORCEUOW, or RESYNC options of **SET IPCONN** and **SET UOW** do not clear resync information. If you want to do this, you must use **SET IPCONN NOTPENDING** or **SET IPCONN NORECOVDATA**.
6. You can issue BACKOUT, COMMIT, FORCEUOW, or RESYNC commands *before* issuing **SET IPCONN NOTPENDING** or **SET IPCONN NORECOVDATA** commands.

CEMT SET IRC

Open or close the interregion communication (IRC) facility.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

New MRO connections can be installed without first closing IRC. However, if you want to modify existing MRO connections, IRC must be closed using the SET IRC CLOSED command. For example:

```
CEMT SET IRC CLOSED
CEDA INSTALL GROUP(groupname)
CEMT SET IRC OPEN
```

Syntax

Press the Clear key to clear the screen. There are two ways to start this transaction:

- Type `CEMT SET IRC` (or suitable abbreviations for the keywords). The resulting display lists the current status, similar to that obtained by using the **CEMT INQUIRE IRC** command. You can tab to the highlighted field and overtype it with the required value.
- Type `CEMT SET IRC` (or suitable abbreviations for the keywords), followed by the attribute setting that you want to change. For example, `cemt s ir o` initializes interregion communication.

Typing `?` at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET IRC

Options

CLOSED
Interregion communication for this CICS system is terminated after all tasks using IRC sessions have terminated normally.

IMMCLOSE
All tasks using IRC sessions are to be abended when they next try to use the session. Interregion communication is terminated when all tasks that were using IRC have been abended or have terminated normally.

OPEN
Interregion communication for this CICS system is initialized. This allows connections to be established between this system and other systems, including DL/I batch regions and non-CICS client programs using the external CICS interface (EXCI). You can control the availability of individual connections with the INSERVICE|OUTSERVICE settings on the connection names.

CEMT SET JOURNALNAME

Enable or disable a CICS user journal.

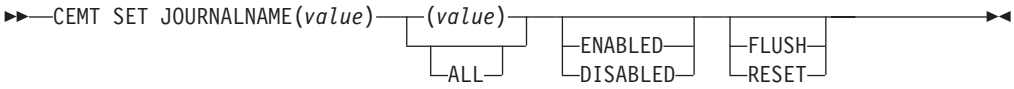
Description

The SET JOURNALNAME command allows you to enable or disable a CICS user journal. It has no effect on a journal that is being used as the forward recovery log or auto-journal for a VSAM file until the next time the file is opened or on the system log.

You can use SET JOURNALNAME for a journal name that is not currently known to CICS. CICS dynamically creates an entry for the specified journal and, if necessary, defines it to the MVS system logger using a matching JOURNALMODEL definition (see the *CICS Resource Definition Guide*.)

Syntax

CEMT SET JOURNALNAME



Options

(*value***)**
Specifies the journal name.

To modify journals defined with a numeric identifier in the range 1–99, specify journal name DFHJ*nn*, where *nn* is the journal number.

You cannot specify DFHLOG or DFHSHUNT, because you are not allowed to modify the status of the system log.

ALL
specifies that any changes you request are made to all resources of the specified type that you are authorized to access.

DISABLED
The journal is disabled. It cannot be used again until it is reenabled by a SET JOURNALNAME(...) STATUS(ENABLED) or ACTION(RESET) command.

ENABLED
The journal is open and is available for use.

FLUSH
The log buffers are written out to the log stream but the journal is not closed.

You can use this option to ensure that all current records are written out to the log stream before the stream is processed using a batch utility.

RESET
The journal is disabled, but can be reenabled by the next SET JOURNALNAME STATUS(ENABLED) or ACTION(RESET) command for this journal.

CEMT SET JVMPOOL

Enable or disable the Java virtual machine (JVM) pool.

In the CICS Explorer, the JVM pools operations view provides a functional equivalent to this command.

Description

You can use the **SET JVMPOOL** command to enable or disable the JVM pool.

The Phaseout, Purge and Forcepurge functions of this command are now deprecated. Use the corresponding functions of the PERFORM JVMPOOL command instead.

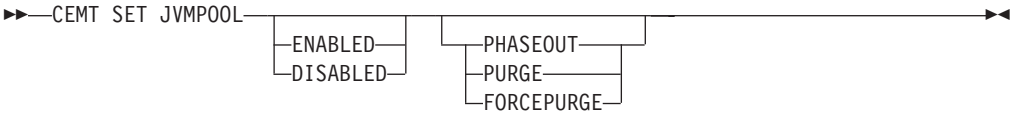
Syntax

Press the Clear key to clear the screen and enter `CEMT SET JVMPOOL` (or suitable abbreviations for the keywords). The resulting display lists the current status,

similar to that obtained by using the **CEMT INQUIRE JVMPOOL** command. You can tab to the highlighted or blank fields and overtype them with the required values.

Typing ? at the end of the first line or the beginning of the second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET JVMPOOL



Options

DISABLED
The pool status is set to disabled, preventing new requests from being serviced from the pool. Programs that were started before the command was issued are allowed to execute to completion.

ENABLED
The pool status is set to enabled for use and Java programs can execute using JVMs from the pool.

FORCEPURGE
All tasks using JVMs are terminated by the SET TASK FORCEPURGE mechanism, and the JVMs are terminated. The shared class cache is deleted when all the JVMs that were using it have been terminated. This function is deprecated; use the CEMT PERFORM JVMPOOL command instead.

PHASEOUT
All JVMs in the pool are marked for deletion. The JVMs are deleted when they finish running their current Java program. The shared class cache is deleted when all the JVMs that were using it have been terminated. This function is deprecated; use the CEMT PERFORM JVMPOOL command instead.

PURGE
All tasks using JVMs in the pool are terminated by the SET TASK PURGE mechanism, and the JVMs are terminated. The shared class cache is deleted when all the JVMs that were using it have been terminated. This function is deprecated; use the CEMT PERFORM JVMPOOL command instead.

CEMT SET JVMSERVER

Change the attributes of a JVM server.

In the CICS Explorer, the JVM servers operations view provides a functional equivalent to this command.

Syntax

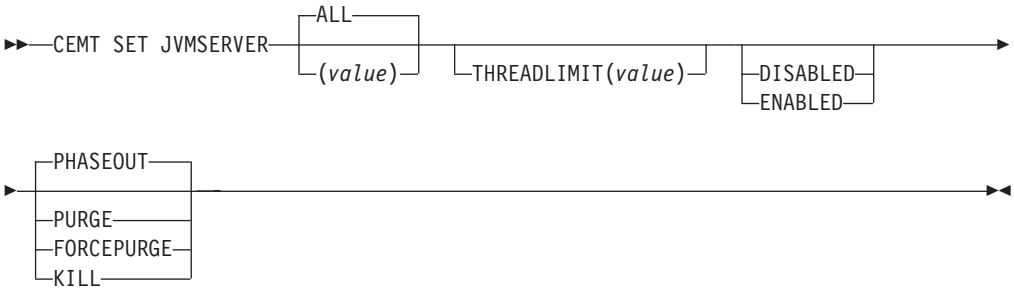
Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET JVMSERVER` followed by one or more JVMSERVER identifiers or ALL. CEMT displays a list of JVMSERVER resources and the current status of each resource. You can then tab to the highlighted or blank fields and overtype them with the required values.

- Type CEMT SET JVMSERVER followed by one or more JVMSERVER identifiers or ALL, followed by the attribute setting that you want to change.

Typing ? at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET JVMSERVER



Options

ALL
Apply any changes to all of the specified resources that you are authorized to access.

DISABLED
Disable one or more JVMSERVER resources. Disabling a JVMSERVER resource shuts down the JVM server and its Language Environment enclave.

ENABLED
Enable one or more JVMSERVER resources. Enabling a JVMSERVER resource initializes the Language Environment enclave and the JVM server.

Note that any changes to JVMProfile and LERUNOPTS are used to update the JVMServer. If you enable a JVMServer after it has been in the DISABLED state, the JVMServer is completely refreshed.

FORCEPURGE
Force purge tasks that are running in the specified JVM server. CICS force purges the tasks by using the **SET TASK FORCEPURGE** command. Any threads running in the JVM are stopped. If the JVMSERVER resource remains in the BEING DISABLED state, some tasks could not be force purged.

Use this option only after you have used the PURGE option where some tasks could not be purged successfully. Data integrity is not guaranteed.

KILL
Terminate tasks that are running in the specified JVM server. CICS stops the tasks by using the **SET TASK KILL** command. Any threads running in the JVM are stopped. The JVMSERVER resource enters the DISABLED state and all work is ended. However, CICS might be left in an unstable state.

Use this option only after you have used the PURGE and FORCEPURGE options where some tasks could not be purged successfully. System and data integrity are not guaranteed. CICS might terminate abnormally.

PHASEOUT
Shut down the JVM server and its Language Environment enclave. All

currently running tasks continue until completion, but no new work is accepted by the JVM server. When all the tasks are finished, the JVMSERVER resource enters the DISABLED state.

PURGE
Purge tasks that are running in the specified JVM server. CICS purges the tasks by using the **SET TASK PURGE** command. Any threads running in the JVM are stopped. CICS purges tasks only when system and data integrity can be maintained. If the JVMSERVER resource remains in the BEING DISABLED state, some tasks could not be purged.

THREADLIMIT(*value***)**
Specifies the maximum number of threads that are available to the threaded system JVM. Each thread runs under a T8 TCB. This value can be in the range 1 - 256 threads. If the thread limit exceeds the maximum of 1024 threads allowed for the CICS region, taking into account the other enabled JVMSERVER resources, CICS adjusts the value accordingly.

value
Specifies the 1 - 8 character name of the JVMSERVER resource.

CEMT SET LINE

Change the status of a line.

Description

You can use the **SET LINE** command to purge or forcepurge a line, to put the line in or out of service, and to adjust the relative line number.

You can specify a list of identifiers (see "List of resource identifiers" on page 116), but you cannot use the asterisk (*) and plus (+) symbols to specify a family of identifiers.

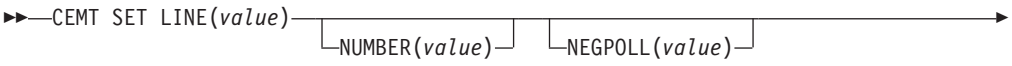
Syntax

Press the Clear key to clear the screen. There are two ways to start this transaction:

- Type `CEMT SET LINE` (or suitable abbreviations for the keywords), followed by a terminal identifier. The resulting display lists the current status, similar to that obtained by using the CEMT INQUIRE LINE command. You can then tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET LINE` (or suitable abbreviations for the keywords), followed by a terminal identifier, then followed by one or more attribute settings that you want to change. For example, `cemt s l(`*value*`) i` makes the line available for use (inservice).

Typing `?` at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET LINE



Options

(*value***)**
Specifies the identifier of a terminal on the line. This request is valid only for non-z/OS Communications Server terminals.

FORCEPURGE
All transactions running with terminals on the line are immediately terminated abnormally. Data integrity is not guaranteed. In some extreme cases (for example, if an error occurs during backout processing), CICS might terminate abnormally.

INSERVICE
The line is available for use. Changing the status of the line does not affect the equivalent status of the terminals on it.

NUMBER (*value***)**
Specifies the relative line number of a pooled line. It is mandatory when inquiring about pooled lines. For information about pooled lines, see the *CICS Resource Definition Guide*.

OUTSERVICE
The line is not available for use. Changing the status of the line does not affect the equivalent status of the terminals on it.

PURGE
Tasks running on terminals associated with the line are abnormally terminated, but task termination occurs only if system and data integrity can be maintained.

CEMT SET MODENAME

Change the status of a mode name.

Description

You can use the **SET MODENAME** command to reset the connection to a remote system or to another CICS region, change the number of sessions available, and acquire or close the sessions associated with the mode name.

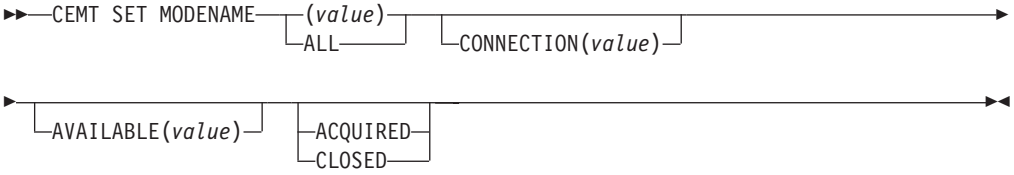
Syntax

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET MODENAME` (or suitable abbreviations for the keywords), followed by either (*modename*) or ALL. The resulting display lists the current status, similar to that obtained by using the **CEMT INQUIRE MODENAME** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET MODENAME` (or suitable abbreviations for the keywords), followed by either (*modename*) or ALL, then followed by one or more attribute settings that you want to change. For example, `cemt s mod(modename) acq` causes CICS to acquire all the negotiated contention-winner sessions in the group named by modename.

Typing ? at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET MODENAME



Options

(*value*)
Specifies one or more names (1 - 8 characters) of a group of sessions. Mode names do not need to be unique. You cannot specify a list of names, but you can use the asterisk (*) and plus (+) symbols to specify a family of mode names.

A mode name that consists of eight blanks is valid. To specify this name, enclose the string of blanks in single quotation marks, for example:

```
MODENAME('   ')
```

ACQUIRED
CICS is to acquire (or bind) all the negotiated contention-winner sessions in the group.

ALL
Specifies that any changes you request are done to all resources of the specified type that you are authorized to access.

AVAILABLE (*value*)
Specifies the maximum number of sessions within the group that can be concurrently allocated for use. This value must be the same for both sides involved in the session.

CLOSED
CICS is to release all sessions in the group. The sessions are quiesced and unbound. No further sessions can be acquired by another CICS system. No further activity is possible on the sessions in the group until the group is reacquired by the same CICS system in which it has been closed.

CONNECTION (*value*)
Specifies an identifier (1 - 4 characters) defined for an IRC or ISC connection.

CEMT SET MONITOR

Change the status of monitoring.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

Use the **SET MONITOR** command to select the classes of monitoring data that you want collected, switch monitoring on or off, and set the status of monitoring options. When you change the status of CICS monitoring from OFF to ON, CICS starts to accumulate monitoring data and to write monitoring records for the active

monitoring classes, for all tasks that start *after* the change is made. The monitoring status of tasks being processed is not affected.

When you change the status of CICS monitoring from ON to OFF, CICS stops writing monitoring data immediately. No monitoring data is accumulated for tasks that start after the change is made. If you turn off monitoring, also set NOPERF to ensure that buffers that contain recorded data for completed tasks are flushed; otherwise, some of this data might be lost.

The monitoring class settings can be changed regardless of whether monitoring is ON or OFF. If monitoring is OFF, you can change the class settings as required in preparation for the next time STATUS is set to ON.

If monitoring is ON, changes to the class settings have the following effects:

- Changing a class from inactive to active has no effect on the data written for running tasks.
- Changing a class from active to inactive prevents monitoring data for that class being written for running tasks as follows:

  PERFORMANCE
  Any performance class data remaining in the output buffer is recorded, and no further data is collected.

  EXCEPTION
  Monitoring data is not written at the completion of the exception condition.

  RESOURCE
  Any transaction resource monitoring data remaining in the output buffer is recorded, and no further data is collected.

  IDENTITY
  Any identity class data remaining in the output buffer is recorded, and no further data is collected.

Exception class, transaction resource class, identity class, and performance class data are gathered during the life of a running transaction. When you change these settings, be aware of their impact on transactions that are running. For example, if a transaction is started when the performance class is set to PERF and monitoring is ON, performance class data is accumulated for that transaction. If you switch the performance class to NOPERF while the transaction is running, and then back to PERF before the transaction ends, performance data will be missing from the final monitoring data for that transaction, even though, when it started and when it ended, the performance class was set to PERF. Transaction resource class and identity class data is written at the end of a task, and is written only if the monitoring class is set (to RESRCE for the transaction resource class, or IDNTY for the identity class) at the point when the task ends.

The default is for monitoring records to be compressed, except for identity class records, which are not compressed. If you change the setting for the data compression option, the new setting applies to all monitoring SMF records written from that point onwards, even if they are for a task being processed at the time the change is made. The new setting also applies to any records that are in the buffer waiting to be written to SMF at the time the change is made. The change applies only until a CICS restart, at which time the setting for the data compression option reverts to that specified in the monitoring control table (MCT) controlled by the DFHMCT macro.

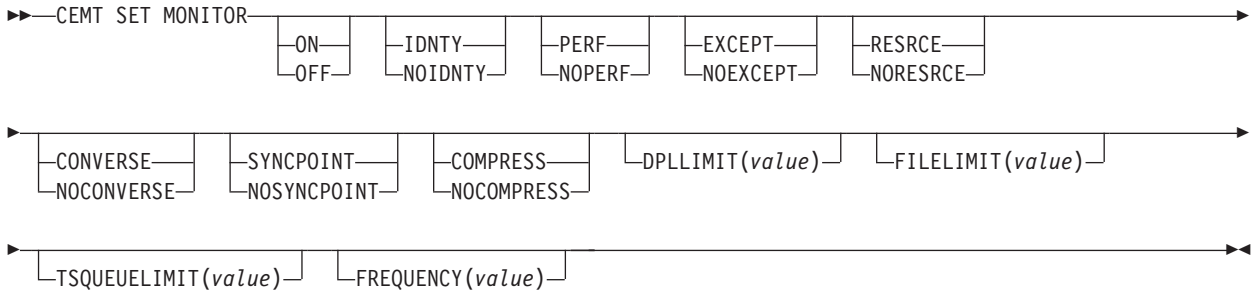
Syntax

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET MONITOR` (or suitable abbreviations for the keywords). The resulting display lists the current status, similar to that obtained by using the **`CEMT INQUIRE MONITOR`** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET MONITOR` (or suitable abbreviations for the keywords), followed by one or more attribute settings that you want to change. For example, `cemt s mon on p ex` sets CICS monitoring ON and collects performance-class and exception-class data.

Typing `?` at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET MONITOR



Options

`COMPRESS`
Data compression is performed for monitoring records. The default is for monitoring records to be compressed.

`CONVERSE`
Conversational tasks have separate performance class records produced for each pair of terminal control I/O requests (Converse or Send/Receive pair).

`DPLLIMIT(`*`value`***`)`**
The maximum number of distributed program link requests for which CICS performs transaction resource monitoring. The value specified must be in the range 0 - 64.

`EXCEPT`
The exception class of monitoring data is collected.

`FILELIMIT(`*`value`***`)`**
The maximum number of files for which CICS performs transaction resource monitoring. The value specified must be in the range 0 - 64.

`FREQUENCY(`*`value`***`)`**
The interval for which monitoring automatically produces a transaction performance class record for any long-running transaction that has been in the system for a greater time than the interval. Frequency times are 0 or in the range 000100 - 024000. The default frequency value is 0, which specifies that

FREQUENCY monitoring is inactive; that is, no performance class records are produced for long-running transactions.

CICS can produce a performance class monitoring record in this way only when the long-running transaction is running on the quasi-reentrant (QR) or concurrent (CO) mode TCBs.

IDNTY
The identity class of monitoring data is to be collected.

NOCOMPRESS
Data compression is not to be performed for monitoring records.

NOCONVERSE
Conversational tasks are not to have separate performance class records produced for each pair of terminal control I/O requests (Converse or Send/Receive pair).

NOEXCEPT
The exception class of monitoring data is not to be collected.

NOIDNTY
The identity class of monitoring data is not to be collected.

NOPERF
The performance class of monitoring data is not to be collected.

NORESRCE
Transaction resource monitoring data is not to be collected.

NOSYNCPOINT
A transaction performance class record is not to be produced when a transaction takes an explicit or implicit sync point (unit-of-work), except when the sync point is part of task termination or a sync point rollback.

OFF
CICS monitoring is switched off. When monitoring is OFF, no monitoring data is accumulated or collected, regardless of the settings of the monitoring data classes.

ON
CICS monitoring is switched on. When monitoring is ON, monitoring data is accumulated continuously and is collected (written out to an SMF data set) for each of the classes of monitoring data that is active.

PERF
The performance class of monitoring data is collected.

RESRCE
Transaction resource monitoring data is collected.

SYNCPOINT
A transaction performance class record is produced when a transaction takes an explicit or implicit sync point (unit-of-work), except when the sync point is part of task termination or a sync point rollback.

TSQUEUELIMIT(*value***)**
The maximum number of temporary storage queues for which CICS performs transaction resource monitoring. The value specified must be in the range 0 - 64.

CEMT SET MQCONN

Change information about the attributes of the connection between CICS and WebSphere MQ, and start or stop the connection.

In the CICS Explorer, the Websphere MQ Connections operations view provides a functional equivalent to this command.

Use the **SET MQCONN** command to change attributes of the currently installed MQCONN resource definition, which defines the connection to WebSphere MQ, and starts and stops the connection.

Because only one MQCONN resource definition can be installed at a time, the name of the MQCONN resource definition is not required on input.

This command does not set the INITQNAME attribute of the MQCONN resource definition, which specifies the name of the initiation queue. If you want to change the MQINI resource definition, you must reinstall the MQCONN resource definition with an appropriate MQINI attribute.

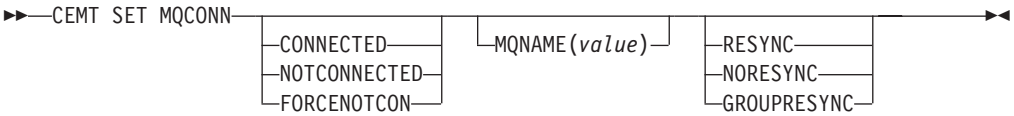
Syntax

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET MQCONN` (or suitable abbreviations for the keywords). The resulting display lists the status, similar to that obtained by using the **CEMT INQUIRE MQCONN** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET MQCONN` (or suitable abbreviations for the keywords), followed by one or more attribute settings that you want to change.

Typing `?` at the beginning of either the first or second line gives a syntax prompt.

SET MQCONN



Conditions: NOTAUTH, NOTFND

Options

CONNECTST(*value*)
Starts or stops the connection between CICS and WebSphere MQ. The possible values are as follows:

CONNECTED
Starts the CICS-WebSphere MQ connection. If the requested queue manager is active, control returns to the application when CICS and WebSphere MQ are connected. If the requested queue manager is not active, CICS connects to WebSphere MQ as soon as the requested queue manager becomes active.

NOTCONNECTED
Stops the CICS-WebSphere MQ connection. Existing transactions are

allowed to finish before the connection is stopped. The request is asynchronous in nature; that is, control is returned to the application before the request is complete.

FORCENOTCON
Stops the CICS-WebSphere MQ connection. Any CICS transactions currently using WebSphere MQ are abnormally stopped, and the connection to WebSphere MQ is stopped. The request is synchronous in nature; that is, control is not returned to the application until the connection is stopped.

MQNAME(*value***)**
Specifies the 1 - 4 character name of a WebSphere MQ queue manager or queue-sharing group to which CICS is to connect. CICS attempts to connect to the queue manager or to any active member of the queue-sharing group. You can change MQNAME only when CICS is not connected to WebSphere MQ.

When you specify MQNAME, the queue manager name or queue-sharing group that you specified in the MQNAME attribute of the installed MQCONN resource definition is replaced with the name that you specified on this command. If you want to revert to the original queue manager or queue-sharing group, you must set MQNAME again.

RESYNCMEMBER(*value***)**
This option applies only if you have specified a queue-sharing group for the CICS-WebSphere MQ connection. It specifies the strategy that CICS adopts if outstanding units of work are being held for the last queue manager to which CICS was connected from the queue-sharing group. Units of work that are shunted indoubt are not included in this process, because CICS itself cannot resolve those units of work at this time. Resynchronization for those UOWs occurs when CICS has resynchronized with its remote coordinator. The possible values are as follows:

RESYNC
CICS connects to the same queue manager.

NORESYNC
CICS makes one attempt to connect to the same queue manager. If that attempt fails, CICS connects to any member of the queue-sharing group and issues a warning about the outstanding units of work.

GROUPRESYNC
CICS connects to any member of the queue-sharing group. The queue manager is chosen by WebSphere MQ and it asks CICS to resolve indoubt units of work on behalf of all eligible queue managers in the queue-sharing group. This function is called *group unit of recovery*. The GROUPRESYNC option can be used only you are when running a release of WebSphere MQ that supports group unit of recovery for CICS and when the GROUPUR attribute has been enabled in the WebSphere MQ queue managers.

When an attempt is made to connect CICS to WebSphere MQ by using an EXEC CICS SET MQCONN CONNECTED command and RESYNCMEMBER(GROUPRESYNC) is set but WebSphere MQ does not support group unit of recovery, or group unit of recovery is not enabled, then WebSphere MQ rejects the connection attempt. The connection attempt results in the SET command failing with INVREQ and RESP2=9 (connection error).

Do not change the settings for RESYNCMEMBER when units of work are outstanding in WebSphere MQ because the units of work cannot then be

resolved. A unit of work held in CICS is identified with a resource manager qualifier. When RESYNCMEMBER(GROUPRESYNC) is used, the qualifier is the name of the queue-sharing group, otherwise the qualifier used is the name of the individual queue manager.

CEMT SET NETNAME

Change the status of networks.

In the CICS Explorer, the Terminals operations view provides a functional equivalent to this command.

Description

If you are familiar with network names, you can use this command as an alternative to the **CEMT SET TERMINAL** command (see "CEMT SET TERMINAL" on page 515). However, you cannot use the **CEMT SET NETNAME** command for APPC sessions.

Netname is the name by which the remote system or terminal is known to the network. The remaining operands for NETNAME are identical to those for TERMINAL.

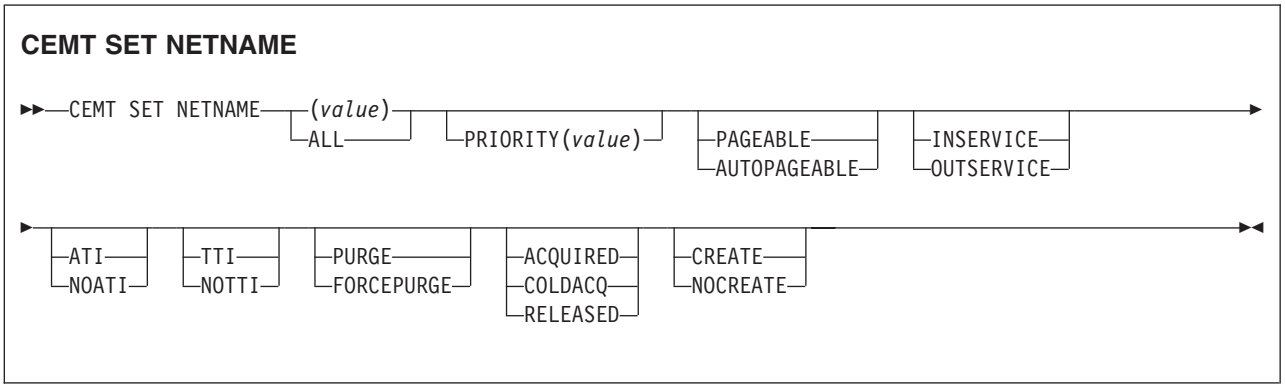
Netnames do not need to be unique, and you cannot specify a list of names.

Syntax

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET NETNAME` (or suitable abbreviations for the keywords), followed by netname or ALL. The resulting display lists the current status, similar to that obtained by using the **CEMT INQUIRE NETNAME** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET NETNAME` (or suitable abbreviations for the keywords), followed by netname or ALL, then followed by one or more attribute settings that you want to change. For example, `cemt s n(netname) i at` resets the named remote system or terminal to in service and available for use.

Typing ? at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.



Options

(*value***)**
Specifies the name by which the remote terminal is known in the network.

ACQUIRED (*z/OS Communications Server only***)**
CICS is in session with the logical unit represented by the terminal.

ALL
Specifies that any changes you request are made to all resources of the specified type that you are authorized to access.

ATI (*z/OS Communications Server only***)**
The terminal is available for use by transactions that are initiated automatically from within CICS or, if the terminal is an ISC session, by transactions that are using this session as an alternative facility to communicate with another system.

AUTOPAGEABLE (*z/OS Communications Server only***)**
Pages after the first in a series are to be written to the terminal automatically.

COLDACQ (*z/OS Communications Server only***)**
This is a special form of ACQUIRED, where no resynchronization is required. If the previous session abended, the use of COLDACQ overrides CICS integrity control. This could lead to integrity problems. Also, you should check the CSMT log for an activity keypoint after the restart of a session following a CICS failure. If there is no activity keypoint, you should issue COLDACQ again after the next emergency restart.

CREATE (*z/OS Communications Server only***)**
If the terminal is not in session, CICS is to acquire it if it is needed to satisfy an ATI request. This cannot be specified for IRC sessions.

FORCEPURGE
All transactions running with this terminal are immediately terminated abnormally. Data integrity is not guaranteed. In some extreme cases (for example, if an error occurs during backout processing), CICS might terminate abnormally.

INSERVICE (*z/OS Communications Server only***)**
The terminal is available for use. For z/OS Communications Server, INSERVICE means that the terminal can be ACQUIRED.

NOATI (*z/OS Communications Server only***)**
The terminal is not available for use by transactions that are initiated automatically from within CICS or, if the terminal is an ISC session, by transactions that are using this session as an alternative facility to communicate with another system.

Note: A terminal cannot be defined with both NOATI and NOTTI.

NOCREATE (*z/OS Communications Server only***)**
If the terminal is not in session, CICS is not to acquire it to satisfy an ATI request. A session must be started by, for example, a logon request or a CEMT SET TERMINAL ACQUIRED command before the ATI request can be satisfied.

If NOCREATE is set for an LU6.1 ISC session and there are no allocatable sessions left, the connection is placed OUTSERVICE.

NOTTI (*z/OS Communications Server only***)**
This terminal cannot be used by transactions.

Note: A terminal cannot be defined with both NOATI and NOTTI.

OUTSERVICE (*z/OS Communications Server only*)
The terminal is not available for use. Setting a terminal OUTSERVICE means that the terminal can no longer be used by transactions. If PURGE or FORCEPURGE is also specified, any transaction using the terminal is terminated abnormally. If PURGE or FORCEPURGE is not specified, the transaction is allowed to terminate normally, but no further transactions are allowed to use the terminal. For z/OS Communications Server, setting a terminal OUTSERVICE also causes it to be released and the operator to be signed off, either immediately or when the current transaction has terminated.

In an LU6.1 ISC session the connection is set OUTSERVICE if there are no allocatable sessions left.

To set an IRC session OUTSERVICE, you must set the connection to the MRO partner OUTSERVICE.

PAGEABLE (*z/OS Communications Server only*)
Pages after the first in a series are to be written to the terminal only if you request them to be.

PRIORITY (*value*) **(z/OS Communications Server only)**
Specifies the priority of a terminal relative to other terminals. The priority of a task is the sum of the transaction priority, the terminal priority, and the operator priority. Priority has no meaning for terminals that are ISC sessions being used as alternative facilities. The value must be in the range 0–255, where 255 is the highest priority.

PURGE
Tasks running on this terminal are abnormally terminated, but task termination occurs only if system integrity can be maintained.

RELEASED (*z/OS Communications Server only*)
CICS is not in session with the logical unit represented by the terminal. Setting a terminal RELEASED causes the session to be terminated. Running transactions are allowed to finish unless PURGE or FORCEPURGE is also specified.

TTI (*z/OS Communications Server only*)
This terminal can be used by transactions.

CEMT SET PIPELINE

Enable or disable a PIPELINE resource.

In the CICS Explorer, the Pipelines operations view provides a functional equivalent to this command.

Description

If the optional attributes are omitted, the command has no effect.

Syntax

Press the Clear key to clear the screen. There are two ways to start this transaction:

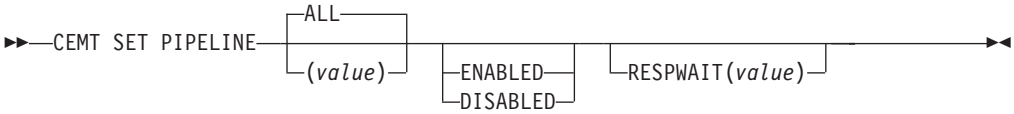
- Type `CEMT SET PIPELINE` (or suitable abbreviations for the keywords), followed by one or more PIPELINE identifiers, or ALL. The resulting display lists the

current status, similar to that obtained by using the **CEMT INQUIRE PIPELINE** command. You can tab to the highlighted or blank fields and overtype them with the required values.

- Type `CEMT SET PIPELINE` (or suitable abbreviations for the keywords), followed by one or more PIPELINE identifiers, or ALL, then followed by one or more attribute settings that you want to change. For example, `cemt s pi al e` makes all PIPELINE resources available for use (enabled).

Typing `?` at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

SET PIPELINE



Conditions: NOTAUTH, NOTFND

Options

ALL
Any changes you request are made to all resources of the specified type that you are authorized to access.

(*value*)
Specifies the 1- to 8-character name of the PIPELINE resource.

ENABLESTATUS
Sets the status of the PIPELINE:

ENABLED
The PIPELINE is ready for use.

DISABLED
The PIPELINE is not processing requests, and cannot accept new work. It might have failed to initialize, or might be explicitly disabled.

RESPWAIT(*value*)
Specifies the time, in seconds, that an application program waits for a response message from a remote web service. The value can range from 0 to 9999 seconds.

If you do not specify a value, the default timeout value of the transport protocol is used.

- The default timeout value for HTTP is 10 seconds.
- The default timeout value for MQ is 60 seconds.

CEMT SET PROCESSTYPE

Change the attributes of a CICS business transaction services process-type.

In the CICS Explorer, the Processtypes operations view provides a functional equivalent to this command.

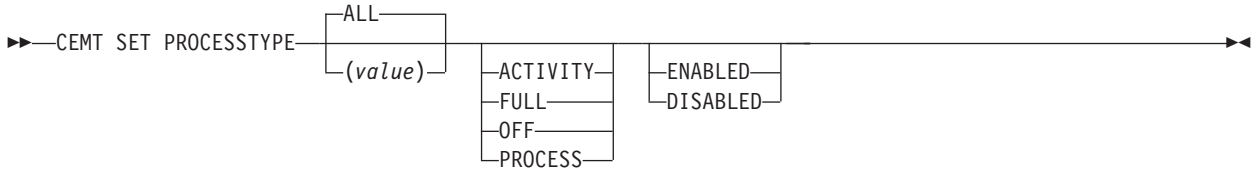
Description

SET PROCESSTYPE enables you to change the current state of audit logging and the enablement status of BTS PROCESSTYPE definitions installed on this CICS region.

Note: Process-types are defined in the process-type table (PTT). CICS uses the entries in this table to maintain its records of processes (and their constituent activities) on external data sets. If you are using BTS in a single CICS region, you can freely use the SET PROCESSTYPE command to modify your process-types. However, if you are using BTS in a CICSPlex, it is strongly recommended that you use CPSM to make such changes. This is because it is essential to keep resource definitions in step with each other, across the CICSPlex.

Syntax

CEMT SET PROCESSTYPE



Options

ACTIVITY|FULL|OFF|PROCESS
specifies the level of audit logging to be applied to processes of this type.

Note: If the AUDITLOG attribute of the installed PROCESSTYPE definition is not set to the name of a CICS journal, an error is returned if you try to specify any value other than OFF.

The values are:

ACTIVITY
Activity-level auditing. Audit records will be written from:

1. The process audit points
2. The activity primary audit points.

FULL Full auditing. Audit records will be written from:

1. The process audit points
2. The activity primary *and* secondary audit points.

OFF No audit trail records will be written.

PROCESS
Process-level auditing. Audit records will be written from the process audit points only.

For details of the records that are written from the process, activity primary, and activity secondary audit points, see *CICS Business Transaction Services*.

ALL
specifies that any changes you specify are made to all process-types that you are authorized to access.

ENABLED|DISABLED
specifies whether new processes of this type can be created. The values are:

DISABLED
The installed definition of the process-type is disabled. New processes of this type cannot be defined.

ENABLED
The installed definition of the process-type is enabled. New processes of this type can be defined.

PROCESSTYPE(value)
specifies the 8-character name of the process-type whose attributes are to be changed.

CEMT SET PROGRAM

Change the attributes of installed programs.

In the CICS Explorer, the Programs operations view provides a functional equivalent to this command.

Description

Only programs that have been defined in the CICS system definition (CSD) file and installed on the running CICS system are accessible through CEMT.

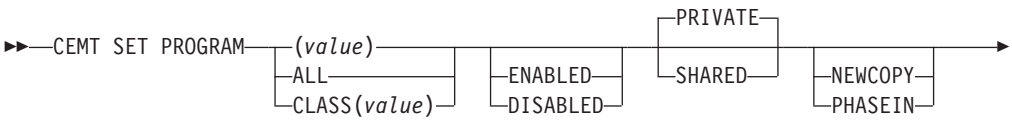
Syntax

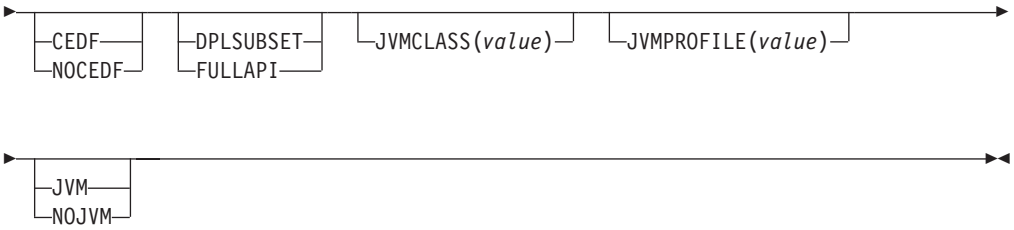
Press the Clear key to clear the screen. There are two ways of starting this transaction:

- Type `CEMT SET PROGRAM` with pgrmid, CLASS(*value*), or ALL. You get a display that lists the current status, similar to that obtained by **`CEMT INQUIRE PROGRAM`**. You can then tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET PROGRAM` followed by one or more attribute settings that you want to change. For example, `cemt s prog(pgrmid) e ne` resets the values for the named program to make it available for use (enabled) and a new copy of the program is used when all the transactions currently using the program have finished (newcopy).

Typing `?` at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET PROGRAM





Options

(*value***)**
Specifies a program identifier (1–8 characters), which is the name of a specific program entry in the table of installed program definitions. If you omit the program name, the ALL option is assumed by default.

You cannot specify the CLASS option and a program name.

ALL
Specifies that any changes you request are to be made to all resources of the specified type that you are authorized to access.

CEDF
When EDF is active, CEDF initiation and termination screens are to be shown by CEDF while this program is running. Other screens are also to be shown unless the program was translated using the NOEDF translator option.

CLASS (*value***)**
Specifies the 2-character suffix of a program list table (PLT). Use this option to inquire on those programs defined in the specified PLT. Note that, unlike a shutdown PLT, it is not mandatory to define a program definition for a startup PLT. However, if you want to use the CLASS option, you must define and install a program resource definition for the PLT specified by *clasid*.

If the *clasid* specified is the suffix of a PLT that does not have a program definition, the **INQUIRE PROGRAM CLASS(clasid)** command returns a `CLASS NOT FOUND` message.

You cannot specify CLASS and a program name.

DISABLED
The program is not available for use. Programs beginning with "DFH" cannot be disabled because these characters are reserved for use by CICS.

DPLSUBSET
The program is restricted to the DPL API subset, as for a distributed program link request, when it runs in the local CICS region.

A program is always restricted to the DPL subset when it is invoked in a remote region via a DPL request, regardless of this option.

ENABLED
The program is available for use.

FULLAPI
The program is not restricted to the DPL subset of the CICS API when it runs in the local CICS region, and can use the full API.

JVM
The program is to run in a pooled Java Virtual Machine (JVM).

JVMCLASS (*value***)**
Specifies the name of the main class in the Java program that is to be given control by the JVM.

JVMPROFILE (*value***)**
Specifies the 8-character name of a JVM profile that is to be used for the pooled JVM in which this Java program runs. Any instances of this program that are currently running in a pooled JVM with the old JVM profile are unaffected, and are allowed to finish running. You cannot enter a JVM profile for Java programs that run in a JVM server because the JVM profile is set on the JVMSERVER resource.

You must enter the profile name using the same combination of uppercase and lowercase characters that are present in the z/OS UNIX file name. If you enter the name of a JVM profile in mixed case when you are using the CEMT transaction, ensure that the terminal has uppercase translation suppressed.

NEWCOPY
CICS is to use a new copy of the program when the program ceases to be in use by any transaction. You can determine whether a module is in use from the RESCOUNT option in an **INQUIRE PROGRAM** command. A value of zero means the program is not in use. It is possible for CICS to replace the program with the new version during a single transaction, at a point when one use of the program has completed, and a subsequent use has yet to start.

CICS loads the new version either from the DFHRPL or dynamic LIBRARY concatenation, or uses an LPA-resident version, depending on the PRIVATE or SHARED options. PRIVATE is the default setting.

You cannot specify NEWCOPY for a program specified with the HOLD option or for a Java program that runs in a JVM. Use the **PERFORM JVMPOOL PHASEOUT** command to refresh Java programs that run in a pooled JVM. Disable and enable the JVMSERVER resource to refresh a Java program that runs in a JVM server.

NOJVM
The program is not to run in a JVM. Any value in the JVMCLASS option is ignored and the runtime environment of the program is unknown until it is next loaded by CICS, at which point its runtime environment is determined.

NOCEDF
All CEDF activities, including initiation and termination screens, are to stop while this program is being processed.

PHASEIN
CICS uses a new copy of the program now for all new transaction requests. CICS continues to use the old copy for all currently running transactions until they have finished (RESCOUNT equal to zero). CICS loads the new version either from the DFHRPL or dynamic LIBRARY concatenation, or uses an LPA-resident version, depending on the PRIVATE or SHARED options. PRIVATE is the default setting.

PHASEIN performs a REFRESH PROGRAM function to inform the loader domain that a new version of the program has been cataloged and that this version of the named program should be used in all future ACQUIRE requests.

Until the program goes to a zero RESCOUNT and a subsequent **ACQUIRE PROGRAM** has been performed, **CEMT INQUIRE PROGRAM ( )** returns information on the first copy of the program.

You cannot specify PHASEIN for a program specified with the HOLD option or for a Java program that runs in a JVM. Use the **PERFORM JVMPOOL PHASEOUT** command to refresh Java programs that run in a pooled JVM. Disable and enable the JVMSERVER resource to refresh a Java program that runs in a JVM server.

PRIVATE
The next new copy of the program is to be loaded from the DFHRPL or dynamic LIBRARY concatenation. This is the default action.

SHARED
CICS uses a shared copy of the program from the LPA as the next new copy of the program. This value is ignored for a Java program that runs in a JVM.

CEMT SET QUEUE

Comments

See "CEMT SET TDQUEUE" on page 512. (QUEUE is accepted for INQUIRE commands, but the correct keyword is TDQUEUE.)

CEMT SET STATISTICS

Change statistics attributes.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

you can use the **SET STATISTICS** command to change some values that control the accumulation and recording of statistics.

For information about printing statistics using the CICS-supplied utility program DFHSTUP, see the *CICS Operations and Utilities Guide*.

NEXTTIME cannot be specified on an INQUIRE or SET STATISTICS command, but its value is displayed when INQUIRE STATISTICS is issued. NEXTTIME indicates the time at which statistics will next be recorded and reset. If SET STATISTICS is in the OFF condition, this is at the end of the day. Otherwise, in the ON condition, it is the earlier time of either the next interval to occur, or the end of the day.

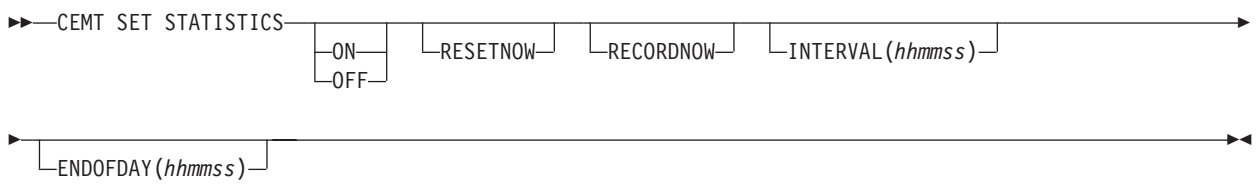
Syntax

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET STATISTICS` (or suitable abbreviations for the keywords). The resulting display lists the current status, similar to that obtained by using the **CEMT INQUIRE STATISTICS** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET STATISTICS` (or suitable abbreviations for the keywords), followed by one or more attribute settings that you want to change. For example, `cemt s st on res` switches CICS statistics ON and resets the statistics counters.

Typing ? at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET STATISTICS



Options

ENDOFDAY (*hhmmss*)
Specifies the end-of-day time for recording statistics. At end-of-day, the current values of the CICS statistics counters are written out to an SMF data set, and the statistics counters are reset.

End-of-day times must be in the range 000000-235959. The default end-of-day time is 12 midnight (000000).

INTERVAL (*hhmmss*)
Specifies the length of time during which the statistics counters are incremented. At the end of each interval, the accumulated statistics are recorded and the statistics counters are reset.

Interval times must be in the range 000100 - 240000. The default interval value is 030000. These interval settings only take place if you set the statistics recording status to ON. You can set the statistics recording status ON or OFF by using the RECORDING option on this command, or by using the STATRCD system initialization parameter.

The arrival of the ENDOFDAY time always causes the current interval to be ended (possibly prematurely) and a new interval to be started. Only end-of-day statistics are recorded at the end-of-day time, even if it coincides exactly with the expiry of an interval.

Changing the ENDOFDAY value has an immediate effect on the times at which INTERVAL statistics are recorded. New interval expiry times are recalibrated in accordance with the new end-of-day time.

When you change the INTERVAL value or the ENDOFDAY value (and also when CICS is initialized), the length of the current (or first) interval is adjusted so that it expires after an integral number of intervals from the end-of-day time.

These rules are illustrated by the following example, where **I** indicates an interval recording, INTERVAL is set at 3-hourly intervals, CICS is initialized at 0815 hours, and **E** indicates the ENDOFDAY setting at 1700 hours.

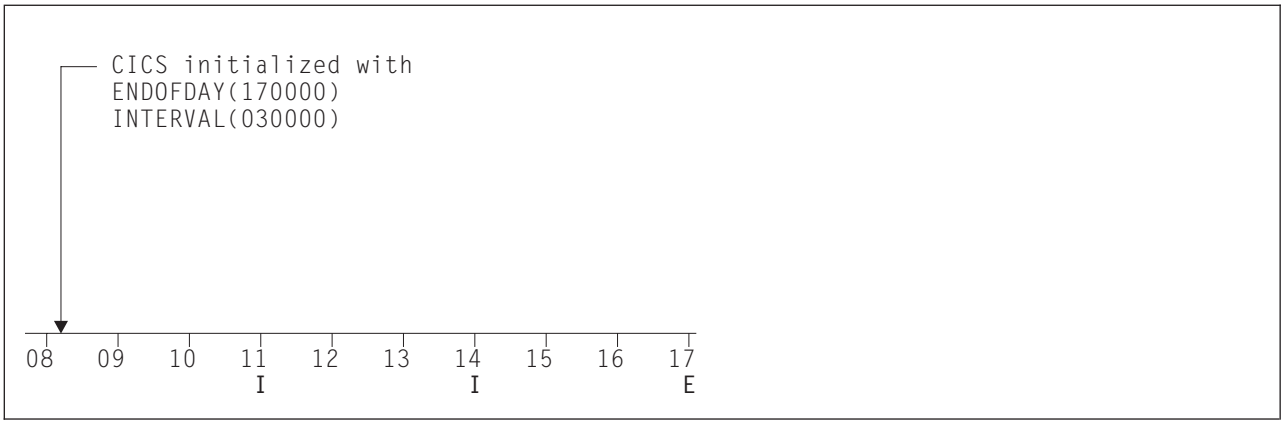


Figure 125. Changing the INTERVAL or ENDOFDAY value

Note: In this example, there would have been an interval at 0800 hours if CICS had been initialized earlier.

If you want your end-of-day recordings to cover 24 hours, set INTERVAL to 240000.

OFF
The recording of interval statistics is switched OFF. End-of-day, requested, and unsolicited statistics are always recorded. Counters are still incremented as usual and the recording of end-of-day, requested, and unsolicited statistics is still made. Statistics can be recorded at any time by issuing a CEMT PERFORM STATISTICS command.

ON
The recording of interval statistics is switched ON. Counters are incremented as usual, and are reset at the expiry of each interval and at end-of-day. Statistics can be recorded at any time by issuing a CEMT PERFORM STATISTICS command.

RECORDNOW
The existing statistics are to be written to an SMF data set. The statistics that you are given are those collected since the last ENDOFDAY or INTERVAL collection, or since the last time the RESETNOW function was used in this or the PERFORM STATISTICS command. RECORDNOW RESETNOW only takes effect when the recording status is changed.

Note: RECORDNOW RESETNOW both resets the statistics counter and also writes the existing statistics to an SMF data set. The statistics that you are given are those collected since the last ENDOFDAY or INTERVAL collection, or since the last time the RESETNOW function was used in this or the PERFORM STATISTICS command. RECORDNOW RESETNOW only takes effect when the recording status is changed.

RESETNOW
The statistics counters are to be reset. The reset takes effect only if the recording option is changed from OFF to ON or from ON to OFF. For information about the effect of reset on each individual counter, see the CICS statistics tables in DFHSTUP reports in the Performance Guide.

The action, for each counter, is one of the following:

- Not reset
- Reset to zero
- Reset to 1

- Reset to current value (this applies to peak values)
- An exception to the above.

CEMT SET SYDUMPCODE

Change the attributes of the system dump codes.

Description

You can use the **SET SYDUMPCODE** command to define the entries that you require in the system dump table. You control system dumps by creating an entry in the system dump table for each dump code that requires a change from the default action.

A system dump code is the CICS message number with the DFH prefix removed. Typically, this leaves a 6-character code comprising 2 alphabetic characters and 4 numeric characters, for example AP0001. The command syntax permits up to 8 characters for the dump code, and the dump codes you specify are padded with trailing spaces.

Each message has its own system default actions that you cannot necessarily override with the SET command. For example, when a serious error occurs, CICS shuts down even if you have asked for no shutdown. See *CICS Messages and Codes Vol 1* for information about the system action for each message.

Other system parameters also take priority over your dump table entries. For example, the EXEC CICS SET SYSTEM NOSYSDUMP command, or a DUMP=NO system initialization parameter, inhibit system dumping. For more information on using dumps, see the *CICS Problem Determination Guide*.

The transaction dump command **SET TRDUMPCODE** can also cause CICS to produce system dumps for the specified transaction dump code.

You can use the SET command to do the following:

- Add a new entry to the system dump table
- Remove an entry from the system dump table
- Indicate whether you want a system dump taken
- Indicate whether the system dump request is local or related
- Indicate whether you want CICS to shut down following the dump
- Set the maximum number of dumps
- Reset the counter of the current number of dumps to zero
- Determine whether a dump produced by CICS is eligible for any suppression by the MVS Dump Analysis and Elimination (DAE) component.

Syntax

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET SYDUMPCODE` (or suitable abbreviations for the keywords), followed by *value* or ALL. The resulting display lists the current status, similar to that obtained by using the **CEMT INQUIRE SYDUMPCODE** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET SYDUMPCODE` (or suitable abbreviations for the keywords), followed by *value* or ALL, then followed by one or more attribute settings that you want to change. For example, the following command adds a system dump code:

```
CEMT SET SYDUMPCODE(value) ADD
```

The following command removes a system dump code:

```
CEMT SET SYDUMPCODE(value) REMOVE
```

The following command changes the maximum number of dump calls for this system dump code to 10:

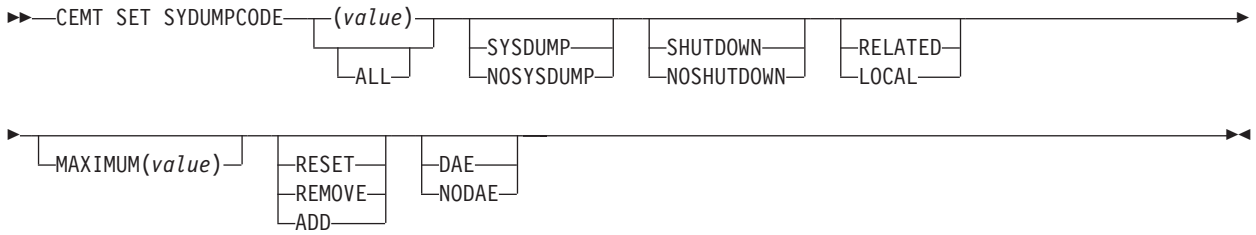
```
CEMT SET SYDUMPCODE(value) MAX(10)
```

The following command shuts down CICS on occurrence of this system dump code:

```
CEMT SET SYDUMPCODE(value) SHUTDOWN
```

Typing ? at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET SYDUMPCODE



Options

(*value*)
Specifies an 8-character system dump code. See *CICS Messages and Codes Vol 1* for details of the codes.

ADD
Add the specified system dump code to the system dump code table.

ALL
Specifies that any changes you request are made to all resources of the specified type that you are authorized to access.

DAE
A system dump is eligible for suppression by the DAE component.

LOCAL
A system dump request is not sent to other MVS images in the sysplex for this system dump code.

MAXIMUM (*value*)
Specifies the maximum number of dump calls for this system dump code that result in system dumps being taken. The value must be in the range 0 - 999. A value of 999 means the default, no limit.

NODAE
A system dump is not suppressed by the DAE component. However, the SUPPRESS and SUPPRESSALL options in the ADYSETxx parmlib member might lead to dump suppression even though NODAE is set here. These options are controlled by the VRADAE and VRANODAE keys in the SDWA. For information about DAE, SUPPRESS, and SUPPRESSALL, see *z/OS MVS Diagnosis: Tools and Service Aids*.

NOSHUTDOWN
CICS is not shut down after the occurrence of an error that corresponds to this dump code.

NOSYSDUMP
Do not produce a system dump for this system dump code.

RELATED
A system dump request is sent to MVS images in the sysplex that are running XCF/MRO connected CICS regions related to the CICS region on which the dump is initiated.

Note: A related CICS region is one in which the unit of work identifiers, in the form of APPC tokens, of one or more tasks match those in the CICS region that issued the dump request.

REMOVE
Remove this system dump code from the system dump code table.

RESET
Reset the current number of calls for this dump code to zero.

SHUTDOWN
Shut down CICS.

SYSDUMP
Produce a system dump for this system dump code.

CEMT SET SYSTEM

Change the system attributes.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

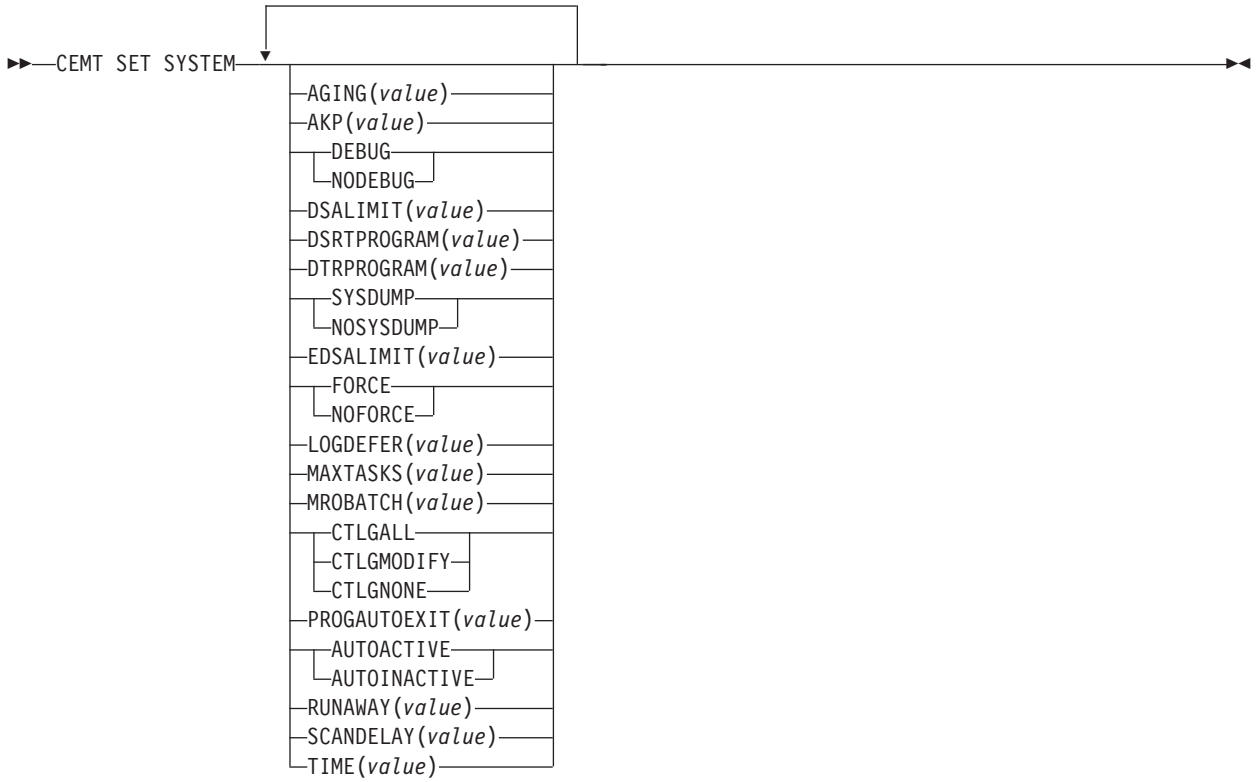
Syntax

Press the Clear key to clear the screen. There are two ways to start this transaction:

- Type `CEMT SET`. You do not need to type the SYSTEM keyword. If you type just `CEMT SET`, SYSTEM is assumed as the default. The current status is displayed, similar to that obtained by using CEMT INQUIRE SYSTEM. You can then tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET` followed by one or more attribute settings that you want to change. For example, `cemt set ag(`*value*`) ma(`*value*`)` resets the values for priority aging and the maximum number of tasks allowed at any one time in the CICS system.

Typing ? at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET SYSTEM



Options

AGING (*value*)
Specifies the time factor for CICS to use in the priority aging algorithm for incrementing the priority of a task. The value is expressed as "milliseconds per unit of priority", and must be in the range 0–65535. The priority aging factor is defined initially by the **PRTYAGE** system initialization parameter, in which the value represents the number of milliseconds that must elapse before the priority of a waiting task can be adjusted upward by 1. For more information about this parameter, see PRTYAGE .

AKP (*value*)
Specifies the activity keypoint trigger value, which is the number of write requests to the CICS system log stream output buffer between the taking of keypoints. The value can be zero, which turns activity keypointing off, or be in the range 200–65535. You cannot change this value if AKPFREQ=0 was specified at system initialization.

AUTOACTIVE
Autoinstall for programs is active. On first use, if a program, map set, or partition set is not defined, the definition is created dynamically.

AUTOINACTIVE
Autoinstall is not active. If a program is not defined, a PGMIDERR or transaction abend occurs when it is referenced.

CTLGALL
All autoinstalled program definitions are to be cataloged and restored on a warm or emergency restart.

CTLGMODIFY
Autoinstalled program definitions are to be cataloged only if they are modified (for example, by a CEMT SET PROGRAM command), so that the modified definitions are restored on a warm or emergency restart.

CTLGNONE
No autoinstalled program definitions are to be cataloged. They are autoinstalled again after a warm or emergency restart.

DEBUG
Specifies that you want to use CICS debugging profiles to select the programs that will run under the control of a debugging tool. The following debugging tools use debugging profiles:

- Debug Tool, for compiled language application programs (programs written in COBOL, PL/I, C, C++, and Assembler)
- Remote debugging tools (for compiled language application programs and Java programs)

Other debugging mechanisms, such as the CICS Execution Diagnostic Facility (CEDF) do not use debugging profiles.

For more information about debugging profiles, see Debugging profiles.

DSALIMIT (*value***)**
Specifies the maximum amount of storage, as a total number of bytes, within which CICS can dynamically allocate storage for the four individual DSAs that reside below 16 MB (below the line). For more information, see DSALIM system initialization parameter in the System Definition Guide. If this parameter specifies a value lower than the current limit, CICS might not implement the new limit immediately, but attempts to do so over time as dynamic storage is freed in the individual DSAs.

Note: Dynamic changes to the DSA limit are cataloged in the *local* catalog, and override values specified in the system initialization table during all forms of restart: initial, old, and warm. The cataloged value is *not* used if:

- You specify startup values as system initialization parameters overrides (for example, in SYSIN).
- You re-initialize the CICS catalog data sets.

DSRTPROGRAM (*value***)**
Specifies the name of the distributed routing program.

DTRPROGRAM (*value***)**
Specifies the name of the dynamic routing program.

EDSALIMIT (*value***)**
Specifies the maximum amount of storage, as a total number of bytes, within which CICS can dynamically allocate storage for the individual DSAs that reside above 16 MB but below 2 GB (above the line). For more information, see EDSALIM system initialization parameter in the System Definition Guide. If **EDSALIMIT** specifies a value lower than the current limit, CICS might not implement the new limit immediately, but attempts to do so over time as dynamic storage is freed in the individual DSAs.

Note: Dynamic changes to the EDSA limit are cataloged in the *local* catalog, and override values specified in the system initialization table during all forms of restart: initial, old, and warm. The cataloged value is *not* used if:

- You specify startup values as system initialization parameters overrides (for example, in SYSIN).
- You re-initialize the CICS catalog data sets.

FORCE
Force all CICSAPI user application programs to run under the QR TCB, even if they are defined with the CONCURRENCY(THREADSAFE) attribute. Force does not apply to certain programs, for example OPENAPI programs, or C or C++ programs compiled with XPLINK. For details, see FORCEQR system initialization parameter in the System Definition Guide.

LOGDEFER (*value*)
Specifies the log deferral interval in milliseconds, which is the period of time used by CICS Log Manager when determining how long to delay a forced journal write request before invoking the MVS system logger. It is in the range 0-65535. For more information, see LGDFINT system initialization parameter in the System Definition Guide.

MAXTASKS (*value*)
Specifies the maximum number of tasks, both active and suspended, allowed at any one time in the CICS system. It must have a value in the range 1–999.

Note: The value assigned to MAXTASKS might be less than the requested value, because of CICS storage constraints. If this occurs, the message CEILING REACHED is displayed when the request is made.

MROBATCH (*value*)
Specifies the number of MRO requests from connected regions that are to be batched before this region is posted. It must have a value in the range 1–255.

NODEBUG
Specifies that you do not want to use CICS debugging profiles to select the programs that will run under the control of a debugging tool. The following debugging tools use debugging profiles:

- Debug Tool, for compiled language application programs (programs written in COBOL, PL/I, C, C++, and Assembler)
- Remote debugging tools (for compiled language application programs and Java programs)

Other debugging mechanisms, such as the CICS Execution Diagnostic Facility (CEDF) do not use debugging profiles.

For more information about debugging profiles, see Debugging profiles.

NOFORCE
Do not force user application programs to run under the QR TCB. CICS is to honor the CONCURRENCY(THREADSAFE) attribute on user application programs, and allow user programs to run on an open TCB to avoid unnecessary TCB switching.

NOSYSDUMP
Specifies that system dumps are to be suppressed.

PROGAUTOEXIT (*value*)
Specifies the name of the user-provided program that is called by the program autoinstall code to select or modify a model definition.

RUNAWAY (*value*)
Specifies the interval, in milliseconds, for which a task can have control before it is assumed to be looping.

You can specify 0, or a value in the range 500–2700000. The value chosen is rounded down to a multiple of 500. CICS purges a task if it has not given up control after this interval (that is, if the task appears to be looping). If you specify zero, runaway task control is inoperative (that is, tasks do not get purged if they appear to be looping). The value is independent of, and can be less than, the value in the TIME operand.

CICS runaway-task detection is based upon task time, that is, the interval is decremented only when the task has control of the processor. You do not, therefore, need to allow for the effect of other jobs when setting the value.

SCANDELAY (*value*)
Specifies the maximum number of milliseconds between the receipt of a z/OS Communications Server request, and the time when CICS begins to process the request. The number must be in the range 0–5000, and must be less than or equal to the TIME value.

In the summary report of the dispatcher statistics, SCANDELAY is referred to as ICVTSD.

SYSDUMP
Specifies that system dumps are not to be suppressed.

TIME (*value*)
Specifies the interval, in milliseconds, for which CICS releases control to the operating system if no transactions are ready to resume processing. This interval is known as the *region exit interval*.

It can have a value in the range 100–3600000, and must be greater than or equal to the SCANDELAY value.

In the summary report of the dispatcher statistics, TIME is referred to as ICV.

CEMT SET TASK

Change the status of one or more tasks to terminate a task or change its priority.

In the CICS Explorer, the Tasks operations view provides a functional equivalent to this command.

Description

You can terminate a task and specify whether to maintain system and data integrity, system integrity only, or neither, before the task is terminated. You can set the priority of a task.

You cannot specify a list of identifiers, nor can you use the asterisk (*) and plus (+) symbols to specify a family of tasks.

Syntax

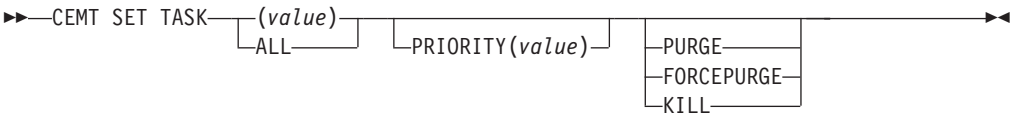
Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET TASK` (or suitable abbreviations for the keywords), followed by (*value*) or ALL. The resulting display lists the current status, similar to that obtained by using the CEMT INQUIRE TASK command. You can then tab to the highlighted or blank fields and overtype them with the required values.

- Type CEMT SET TASK (or suitable abbreviations for the keywords), followed by (*value*) or ALL, then followed by one or more attribute settings that you want to change. For example, cemt s ta(*value*) pu makes the specified task purgeable in system-stall conditions.

Typing ? at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET TASK



Options

(*value*)
The CICS-generated task number. This is in the range 1 - 99999.

ALL
Any changes you request are made to all resources of the specified type that you are authorized to access.

FORCEPURGE
Terminate the task as soon as it is consistent with system integrity, and without regard to data integrity. CICS cannot always determine whether a forced purge is safe; it is possible to abend the system when you specify this option. For example, if a task is forcepurged during backout processing, CICS terminates abnormally. If you want to terminate a task, but do not want to terminate CICS, use the PURGE option instead.

KILL
Terminate the task. System and data integrity is not guaranteed. The KILL option extends the PURGE and FORCEPURGE options. Use this option only after first attempting a purge or forced purge. The KILL option does not guarantee integrity of any kind, but in some situations you can use it to free up a stalled region, thus enabling the region to continue processing. In some cases, for example, if a task is killed during backout processing, CICS terminates abnormally.

PRIORITY (*value*)
Specifies the priority of the task in the range 0 - 255, where 255 is the highest priority.

PURGE
Terminate the task as soon as both system and data integrity can be maintained.

If the task is associated with an open TCB (for example, it involves a Java program running in a JVM on a J8 or J9 mode TCB), you might experience a delay of about a minute before the task is finally purged.

CEMT SET TCLASS

Reset the maximum number of tasks and the purge threshold for a transaction class.

In the CICS Explorer, the Transaction classes operations view provides a functional equivalent to this command.

Syntax

Press the Clear key to clear the screen. There are two ways to start this transaction:

- Type `CEMT SET TCLASS` (or suitable abbreviations for the keywords), followed by (*value*) or ALL. The resulting display lists the current status, similar to that obtained by using the **CEMT INQUIRE TCLASS** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET TCLASS` (or suitable abbreviations for the keywords), followed by one or more attribute settings that you want to change. For example, `cemt s tc(value) p(3)` resets the purge threshold value to 3.

Typing `?` at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET TCLASS

Options

(*value***)**
Specifies the 8-character transaction class name. If no transaction class name is assigned, it defaults to DFHTCL00.

ALL
Specifies that any changes you request are made to all resources of the specified type that you are authorized to access.

MAXACTIVE (*value***)**
Specifies the largest number of transactions in the transaction class that are allowed to run concurrently. The value can be in the range 0-999.

PURGETHRESH (*value***)**
Specifies the limit at which the queuing transactions are purged for the transaction class. It can have a value in the range 0–1000000. Zero means that the transactions are not purged. A nonzero number “n” means that the first (n-1) transactions to arrive are queued and the nth is purged.

CEMT SET TCPIP

Open or close the CICS TCP/IP service.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Description

You can use the **SET TCPIP** command to open or close CICS internal sockets support.

Note: This command has no effect on the sockets support provided by the TCP/IP for z/OS CICS Sockets Feature.

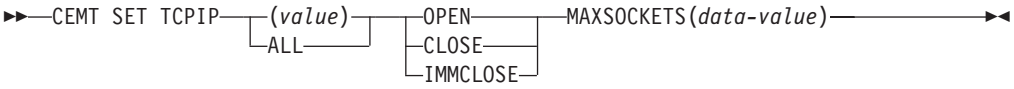
Syntax

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET TCPIP` (or suitable abbreviations for the keywords). The resulting display lists the current status, similar to that obtained by using the **CEMT INQUIRE TCPIP** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET TCPIP` (or suitable abbreviations for the keywords), followed by (*value*) or ALL, then followed by one or more attribute settings that you want to change. For example, `cemt s tcpip status open` opens CICS internal sockets support.

Typing `?` at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET TCPIP



Options

OPENSTATUS

Specifies whether to open or close TCP/IP support, and if TCP/IP support is closed, how the close is done.

OPEN CICS internal sockets support is opened (that is, new incoming work can be processed, and ongoing work can complete).

Note: If you use CEMT SET TCPIP OPEN to initialize CICS sockets, you must also set TCPIP=YES at CICS initialization by SIT or a SIT override. Otherwise, "NOT IN SYSTEM" is returned.

CLOSE

CICS internal sockets support is closed. If it is currently open, CICS quiesces all internal sockets activity and then closes any sockets on which CICS is listening for incoming work. Tasks that are using CICS internal sockets are allowed to complete.

IMMCLOSE

CICS internal sockets support is closed. If it is currently open, any tasks that using it are abnormally terminated, and then CICS closes the socket on which CICS is listening for incoming work.

MAXSOCKETS(*data-value***)**

Specifies the maximum number of IP sockets that the CICS sockets domain can manage.

If the userid under which the CICS job is running has superuser authority, the sockets domain can manage up to 65535 sockets.

If the userid under which the CICS job is running does not have superuser authority, the maximum number of sockets that the sockets domain can manage is limited to the number specified in the **MAXFILEPROC** parameter in SYS1.PARMLIB member BPXPRMxx. If you use the **SET SYSTEM** command to

specify a value greater than that in **MAXFILEPROC**, CICS resets the limit to the value of **MAXFILEPROC**. The message "MAXFILEPROC LIMIT REACHED" is displayed when the request is made.

Note that sockets created by Java programs running on threads that are not managed by CICS do not count towards the limit.

If you reduce the limit to less than the number of sockets currently active, CICS prevents new sockets from being created until the number of active sockets falls below the limit.

CEMT SET TCPIPSERVICE

Modify the status or attributes of a service that uses CICS internal TCP/IP support.

In the CICS Explorer, the TCP/IP services operations view provides a functional equivalent to this command.

Description

You can use the **SET TCPIPSERVICE** command to update information that relates to a service that uses CICS internal TCP/IP support.

Note: This command has no effect on the sockets support provided by the TCP/IP for CICS sockets feature.

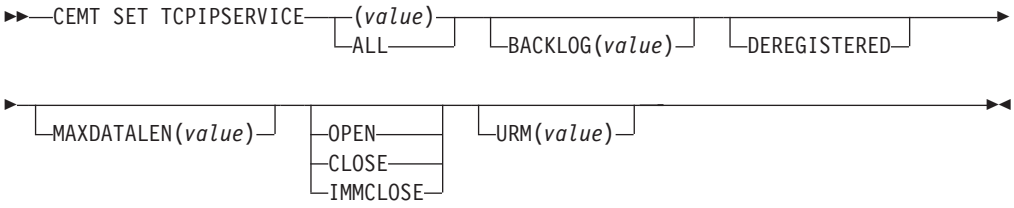
Syntax

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET TCPIPSERVICE` (or suitable abbreviations for the keywords). The resulting display lists the current status, similar to that obtained by using the CEMT INQUIRE TCPIPSERVICE command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET TCPIPS` (or suitable abbreviations for the keywords), followed by (*value*) or ALL, then followed by one or more attribute settings that you want to change. For example, `cemt s tcpips status open` opens CICS internal sockets support.

Typing `?` at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET TCPIPSERVICE



Options

BACKLOG

The maximum number of requests that can be queued in TCP/IP waiting to be

processed by the service. Before you can change this value, the status of the service must be closed. If you set BACKLOG to zero, CICS does not receive any connections.

DNSSTATUS
The current state of Workload Manager (WLM)/Domain Name System (DNS) registration of this TCPIPSERVICE. You can change this status independently from changing the open or closed status of the service.

To account for timing delays in the deregister request reaching the WLM, and the DNS updating its tables, it is advisable to deregister a service before setting it closed. This ensures that client applications do not encounter "Connection Refused" situations during the time between the deregister call being issued and the DNS server updating its tables.

DEREGISTERED
CICS deregisters the group name specified by the DNSGROUP attribute of this TCPIPSERVICE. The WLM macro IWMSRDRS is called and CICS is longer a part of the DNS connection optimization. Any other TCPIPSERVICES that are in the same group (that is, that share the same DNSGROUP attribute) are also deregistered.

MAXDATALEN
The maximum length of data that can be received by CICS as an HTTP server. This can be 3 - 524288K. The default is 32K.

OPENSTATUS
The status of the service:

OPEN CICS internal TCP/IP sockets support starts listening for incoming requests on the specified port.

CLOSE
CICS internal TCP/IP sockets support stops listening for incoming work on the specified port. All active tasks that are using this service are allowed to complete.

IMMCLOSE
CICS internal sockets support stops listening for incoming work on the specified port immediately. Any active tasks that are using this service are terminated abnormally.

URM
Specifies the 8-character name of the program to be used as the service user-replaceable module by the attached task. You can specify either an installation-pecific program or the CICS-supplied default for the service.

CEMT SET TDQUEUE

Changes the attributes of transient data queues.

In the CICS Explorer, the TD Queues operations view provides a functional equivalent to this command.

Description

The **SET TDQUEUE** command changes some of the attributes of a transient data queue. The queue must not be REMOTE or INDIRECT.

You cannot alter the following attributes unless the queue is fully disabled:

- ATIFACILITY
- ATITERMID
- ATITRANID

To disable a transient data destination, the queue must not currently be in use. If it is in use, the queue enters a “disable pending” state. The last unit of work (UOW) to use the queue fully disables it. You can alter the TRIGGERLEVEL, OPENSTATUS, and ENABLESTATUS attributes regardless of whether the queue is enabled or disabled. You cannot alter the value of the ENABLESTATUS attribute while a queue is in a “disable pending” state.

A transient data queue cannot be disabled while it is in use, or while tasks are waiting to use it.

Indirect and remote queues can be disabled at any time because they have no concept of being in use.

If tasks are waiting to use an extrapartition queue, a physically recoverable queue, or a nonrecoverable intrapartition queue and an attempt is made to disable the queue, the queue enters a “disable pending” state. The last task to use the extrapartition queue fully disables it.

If an attempt is made to disable a logically recoverable intrapartition TD queue when there are UOWs enqueued upon it, the queue enters a “disable pending” state. The last UOW to obtain the enqueue fully disables the queue. If a UOW has updated a logically recoverable queue and suffers an indoubt failure, the queue cannot be disabled until the indoubt failure has been resolved.

If a UOW owns an enqueue on a queue that is in a “disable pending” state, it is allowed to continue making updates.

When a queue is in a “disable pending” state, no new tasks can alter the queue's state or its contents. A disabled response is returned when a READQ, WRITEQ, or DELETEQ request is issued against a destination that is in a “disable pending” state.

If a task updates a logically recoverable transient data queue, and attempts to disable the queue and alter an attribute of the queue (for example, ATITRANID) in the same UOW, the call fails. This is because the UOW is a user of the queue, and the queue enters a “disable pending” state. The SET operation on the queue attribute, in this case ATITRANID, fails. The queue does not become fully disabled until the UOW commits or backs out at syncpoint. You are recommended to issue an **EXEC CICS SYNCPOINT** command before you use the **SET TDQUEUE** command to update the queue attribute (ATITRANID).

Syntax

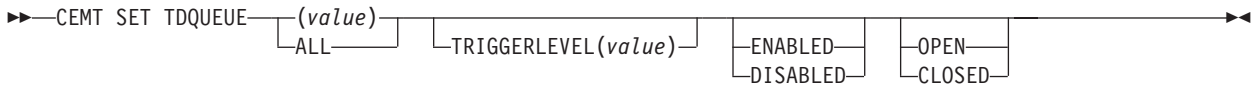
Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET TDQUEUE` (or suitable abbreviations for the keywords), followed by (*value*) or `ALL`. The resulting display lists the current status, similar to that obtained by using the **CEMT INQUIRE TDQUEUE** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET TDQUEUE` (or suitable abbreviations for the keywords), followed by (*value*) or `ALL`, then followed by one or more attribute settings that you wish to

change. For example, `cemt s td(value) en op` makes a named extrapartition queue accessible by applications and open.

Typing `?` at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET TDQUEUE



Options

(*value*)
The 1- to 4-character identifier of a transient data queue.

ALL
Specifies that any changes you request are made to all resources of the specified type that you are authorized to access.

CLOSED
The queue is closed.

DISABLED
The queue cannot be accessed by applications, although it can still be open. A queue that has had an indoubt failure cannot be disabled.

ENABLED
The queue can be accessed by applications.

OPEN
The extrapartition queue is open.

TRIGGERLEVEL (*value*)
Specifies the number of requests for output to a queue that must occur before automatic transaction initiation (ATI) occurs. The value can be 0 - 32767. A value of zero disables ATI.

CEMT SET TEMPSTORAGE

Set the amount of storage that is available to main temporary storage queues in the CICS region.

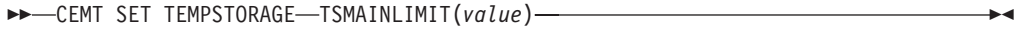
Description

The CEMT SET TEMPSTORAGE command changes the limit for the amount of storage that is available for main temporary storage queues to use.

When you change this limit, check your current setting for the z/OS parameter **MEMLIMIT**. **MEMLIMIT** limits the amount of 64-bit storage that the CICS address space can use. Your setting for **TSMAINLIMIT** must not be greater than 25% of the **MEMLIMIT** value. Use the CEMT command **INQUIRE DSAS** or **INQUIRE SYSTEM** to find the value of **MEMLIMIT** that currently applies to the CICS system.

Syntax

CEMT SET TEMPSTORAGE



Options

TSMAINLIMIT(*value***)**
Specifies the maximum amount of storage that CICS makes available for main temporary storage queues to use. The minimum value is 1048576 bytes (1 MB), and the maximum value is 34359738368 bytes (32768 MB or 32 GB).

You can enter a value in bytes, megabytes followed by the letter M, or gigabytes followed by the letter G. If you enter a value in bytes, it is rounded down to the nearest megabyte.

If you decrease the TSMAINLIMIT setting, CICS attempts to maintain at least 25% free space in allowed storage above current utilization, so that temporary storage write requests do not reach TSMAINLIMIT too rapidly. The value is set as follows:

- If there is currently less than 25% free space, TSMAINLIMIT remains unchanged.
- If at least 25% of the new limit will be free space, the setting is decreased to the value that you choose.
- If less than 25% of the new limit would be free space, the setting is decreased to the current utilization plus 33% of that utilization.

If you increase the TSMAINLIMIT setting, the value is set as follows:

- If the new value is not greater than 25% of the value of the z/OS parameter **MEMLIMIT**, the value that you choose is set.
- If the new value is greater than 25% of the **MEMLIMIT** value, TSMAINLIMIT remains unchanged.

CEMT SET TERMINAL

Change the attributes of named terminals.

In the CICS Explorer, the Terminals operations view provides a functional equivalent to this command.

Description

The **SET TERMINAL** command changes some of the values of a named terminal. It cannot be used for APPC sessions.

In an SNA environment, a terminal is the CICS representation of an SNA logical unit with which CICS can be in communication. (SNA environment means that CICS is using z/OS Communications Server as the access method for communication with SNA logical units.)

This logical unit can be:

- A physical terminal (such as a 3277)
- A function of a cluster controller (such as a 3790 Type 2 batch function)

- An intersystem communication (ISC) or interregion communication (IRC) session that has been generated by the CEDA transaction.

If you are familiar with network names, you may prefer to use the command that uses these names. See "CEMT SET NETNAME" on page 490.

The message "TEMP OUT SERVICE" might be displayed if recovery is in progress for the terminal. To reset the "TEMP OUT SERVICE" indicator, refresh the screen, or overtype the INS field with `INS` or `OUT`. If "TEMP OUT SERVICE" continues to be displayed, investigate the cause of this indicator.

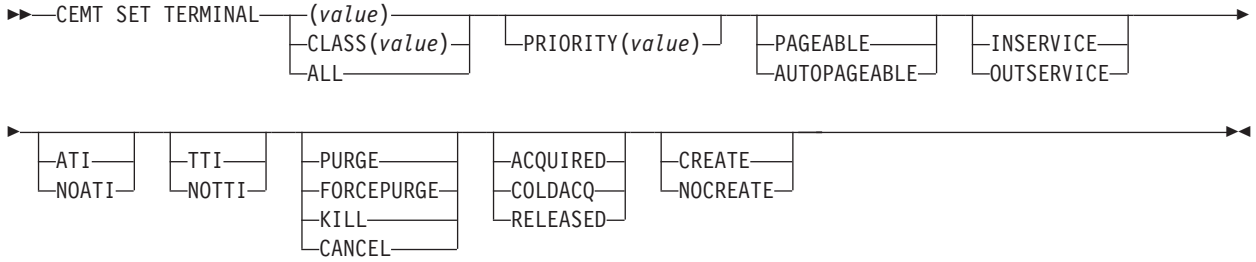
Syntax

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET TERMINAL` (or suitable abbreviations for the keywords), followed by (*value*), CLASS(*value*), or ALL. The resulting display lists the current status, similar to that obtained by using the **CEMT INQUIRE TERMINAL** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET TERMINAL` (or suitable abbreviations for the keywords), followed by (*value*), CLASS(*value*), or ALL, then followed by one or more attribute settings that you want to change. For example, `cemt s te(value) i cr` specifies that a named terminal is available for use (ins ervice) and that it can be acquired automatically by ATI transactions (create).

Typing ? at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET TERMINAL



Options

(*value*)
A terminal identifier (1 - 4 characters) as specified in an installed terminal definition.

For example, if the terminal name is S201, this option is as follows:

```
CEMT SET TERMINAL(S201)
```

ACQUIRED (z/OS Communications Server only)
CICS is in session with the logical unit represented by the terminal.

ALL
Any changes you request are made to all resources of the specified type that you are authorized to access.

ATI (z/OS Communications Server only)
The terminal is available for use by transactions that are initiated automatically from within CICS or, if the terminal is an ISC session, by transactions that are using this session as an alternative facility to communicate with another system.

AUTOPAGEABLE (z/OS Communications Server only)
Pages after the first in a series are to be written to the terminal automatically.

CANCEL
Automatic initiate descriptor (AIDs) queuing for the specified terminal are canceled. AIDs representing scheduled and allocated requests waiting in the local CICS system for the specified terminal are canceled. However, TD AIDs with an associated triggered task already started are not canceled.

When a canceled scheduled request has a precursor in a remote CICS system, this remote AID is canceled asynchronously. Message DFHTF0100 is written to CSMT to indicate how many AIDs have been deleted for the terminal and how many remain.

An "AIDS CANCELED" message appears on the CEMT panel whenever AIDs are deleted using the CANCEL option of the CEMT SET TERMINAL command.

CLASS (*value*)
The 1- or 2-character suffix of a terminal list table (TLT).

COLDACQ (z/OS Communications Server only)
This is a special form of ACQUIRED, where no resynchronization is required. If the previous session abended, the use of COLDACQ overrides CICS integrity control. This could lead to integrity problems. Also, you should check the CSMT log for an activity keypoint after the restart of a session following a CICS failure. If there is no activity keypoint, you should issue COLDACQ again after the next emergency restart.

CREATE (z/OS Communications Server only)
If the terminal is not in session, CICS acquires it if it is needed to satisfy an ATI request.

FORCEPURGE
Any transaction running with this terminal is immediately terminated abnormally. Data integrity is not guaranteed. In some extreme cases (for example, if an error occurs during backout processing), CICS might terminate abnormally.

INSERVICE
The terminal is available for use. For z/OS Communications Server, INSERVICE means that the terminal can be ACQUIRED.

KILL
Terminate the task. System and data integrity is not guaranteed. The KILL option extends the PURGE and FORCEPURGE options. Use this option only after first attempting a purge or forced purge. The KILL option does not guarantee integrity of any kind, but in some situations you can use it to free up a stalled region, thus enabling the region to continue processing. In some cases, for example, if a task is killed during backout processing, CICS terminates abnormally.

NOATI (z/OS Communications Server only)
The terminal is not available for use by transactions that are initiated

automatically from within CICS or, if the terminal is an ISC session, by transactions that are using this session as an alternative facility to communicate with another system.

Note: A terminal cannot be defined with both NOATI and NOTTI.

NOCREATE (z/OS Communications Server only)
If the terminal is not in session, CICS does not acquire it to satisfy an ATI request. A session must be started by, for example, a logon request or a **CEMT SET TERMINAL ACQUIRED** command before the ATI request can be satisfied.

If NOCREATE is set for an LU6.1 ISC session and there are no allocatable sessions left, the connection is placed OUTSERVICE.

NOTTI (z/OS Communications Server only)
This terminal cannot be used by transactions that are initiated from this terminal.

Note: A terminal cannot be defined with both NOATI and NOTTI.

OUTSERVICE
The terminal is not available for use. Setting a terminal OUTSERVICE means that the terminal can no longer be used by transactions. If PURGE or FORCEPURGE is also specified, any transaction using the terminal is terminated abnormally. If PURGE or FORCEPURGE is not specified, the transaction can terminate normally, but no further transactions can use the terminal. For z/OS Communications Server, setting a terminal OUTSERVICE also causes it to be released and the operator to be signed off, either immediately or when the current transaction has terminated.

In an LU6.1 ISC session the connection is set OUTSERVICE if there are no allocatable sessions left.

To set an IRC session OUTSERVICE, you must set the connection to the MRO partner OUTSERVICE.

PAGEABLE (z/OS Communications Server only)
Pages after the first in a series are to be written to the terminal only if you request them to be.

PRIORITY (z/OS Communications Server only)
Specifies the priority of a terminal relative to other terminals. The priority of a task is the sum of the transaction priority, the terminal priority, and the operator priority. Priority has no meaning for terminals that are ISC sessions being used as alternative facilities. The value must be in the range 0 - 255, where 255 is the highest priority.

PURGE
Any transaction running with this terminal is purged only if system and data integrity can be maintained.

RELEASED (z/OS Communications Server only)
CICS is not in session with the logical unit represented by the terminal. Setting a terminal RELEASED causes the session to be terminated. Running transactions are allowed to finish unless PURGE or FORCEPURGE is also specified.

TTI (z/OS Communications Server only)
This terminal can be used by transactions.

CEMT SET TRANSACTION

Change the attributes of a selected transaction.

In the CICS Explorer, the Transactions operations view provides a functional equivalent to this command.

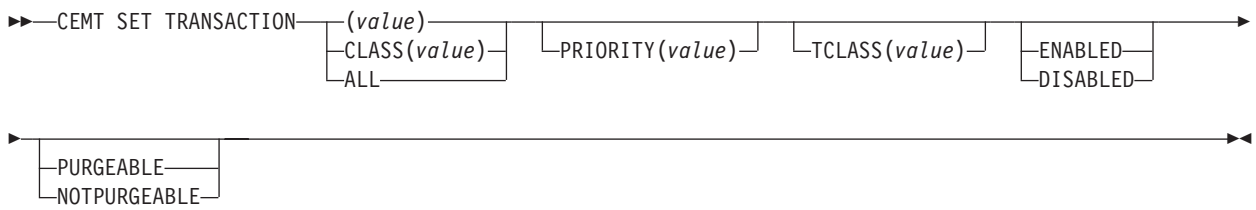
Syntax

Press the Clear key to clear the screen. There are two ways to start this transaction:

- Type `CEMT SET TRANSACTION` (or suitable abbreviations for the keywords), followed by (*value*), CLASS(*value*), or ALL. The resulting display lists the current status, similar to that obtained by using the CEMT INQUIRE TRANSACTION command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET TRANSACTION` (or suitable abbreviations for the keywords), followed by one or more attribute settings that you want to change. For example, `cemt s trans(value) e pu` specifies that a named transaction is available for use (enabled) and can be purged by the system.

Typing ? at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET TRANSACTION



Options

(*value*)
Specifies a transaction identifier (1-4 characters). Only transactions that have been defined in the CICS system definition (CSD) file and installed on the running CICS system are accessible through CEMT.

ALL
Specifies that any changes you request are made to all resources of the specified type that you are authorized to access.

CLASS (*value*)
Specifies the 2-character suffix of a transaction list table (XLT).

DISABLED
The transaction is not available for use. Transactions that have identifiers that begin with the character C cannot be disabled, because these transactions are reserved for use by CICS.

When the CEMT SET TRANSACTION DISABLED command is used, existing transactions run to completion before they become unavailable for use.

Note: If a transaction is disabled, a START command that names this transaction can still be shipped to a remote region. When a task is attached for the requested transaction, CICS checks that the transaction is enabled in the remote region.

ENABLED
The transaction is available for use.

NOTPURGEABLE
The transaction cannot be purged.

PRIORITY (*value***)**
Specifies the priority of a transaction relative to other transactions. When a transaction is running as a CICS task, the priority of a task is the sum of the transaction priority, the terminal priority, and the operator priority. The value must be in the range 0–255, where 255 is the highest priority.

PURGEABLE
The transaction can be purged by the system. This operand relates to the SPURGE attribute on the transaction resource definition and indicates that CICS can purge the transaction in a deadlock timeout situation. See in the *CICS Resource Definition Guide* for information about the SPURGE and DTIMOUT attributes on a transaction resource definition.

TCLASS (*value***)**
Specifies the 8-character transaction class name to which the transaction belongs. If the transaction does not belong to a class, DFHTCL00 is returned. To remove a transaction from its transaction class, set the TCLASS field to DFHTCL00. A series of blanks can be inserted, in which case, DFHTCL00 is returned. An added or changed transaction class **must** be defined.

CEMT SET TRDUMPCODE

Change the status of a transaction dump.

Description

You can use the **SET TRDUMPCODE** command to change entries in the transaction dump table to request a transaction dump. You can also set the maximum number of dumps that can be taken. See the *CICS Problem Determination Guide* for information on using dumps.

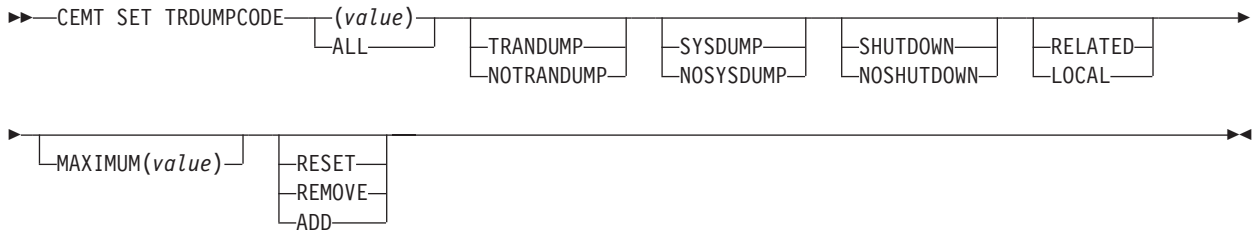
Syntax

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET TRDUMPCODE` (or suitable abbreviations for the keywords), followed by (*value*) or ALL. The resulting display lists the current status, similar to that obtained by using the **CEMT INQUIRE TRDUMPCODE** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET TRDUMPCODE` (or suitable abbreviations for the keywords), followed by one or more attribute settings that you want to change. For example, `cemt s trd(`*value*`) t sh` specifies that a transaction dump is produced for the named transaction dump code and that the CICS system shuts down after an error that corresponds to this dump code occurs.

Typing ? at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET TRDUMPCODE



Options

(*value*)
Specifies a 4-byte transaction dump code. See *CICS Messages and Codes Vol 1* for a list of transaction dump codes.

ADD
Add the specified transaction dump code to the dump code table.

ALL
Specifies that any changes you request are made to all resources of the specified type that you are authorized to access.

LOCAL
Do not send a system dump request to other MVS images in the sysplex that are running XCF/MRO connected CICS regions related to the CICS region on which the dump is initiated.

Note: A related CICS region is one in which the unit of work identifiers, in the form of APPC tokens, of one or more tasks match those in the CICS region that issued the dump request.

MAXIMUM (*value*)
Specifies the largest number of dump calls for this dump code that result in dumps being taken. The range is 0 - 999, although a value of 999 means no limit.

NOSHUTDOWN
Do not shut down the CICS system.

NOSYSDUMP
Do not produce a system dump.

NOTRANDUMP
Do not produce a transaction dump.

RELATED
Send a system dump request to MVS images in the sysplex that are running XCF/MRO connected CICS regions related to the CICS region on which the dump is initiated.

Note: A related CICS region is one in which the unit of work identifiers, in the form of APPC tokens, of one or more tasks match those in the CICS region that issued the dump request.

REMOVE
Remove this code from the dump code table.

RESET
Reset the current count of dump calls for this dump code to zero.

SHUTDOWN
Shut down the CICS system an error that corresponds to this dump code occurs.

SYSDUMP
Produce a system dump.

TRANDUMP
Produce a transaction dump.

CEMT SET TSQUEUE / TSQNAME

Delete a temporary storage queue (TS queue). This topic applies also to the alternative command, **CEMT SET TSQNAME**. In CEMT you can use either command to operate on temporary storage queues (TS queues) with names up to 16 characters long.

In the CICS Explorer, the TS queues operations view provides a functional equivalent to this command.

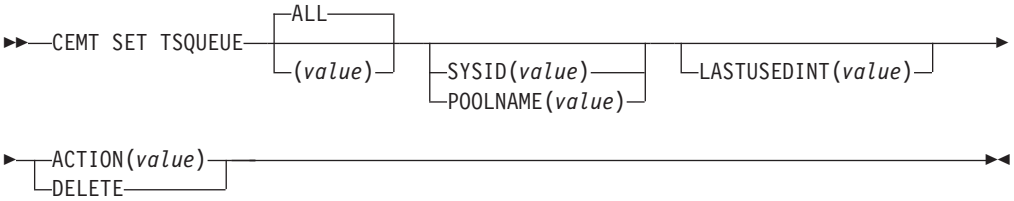
Description

The SET TSQUEUE command enables you to delete a temporary storage queue (TS queue).

The LASTUSEDINT option may be used to ensure that the queue to be deleted has not been referenced since a previous INQUIRE was issued. It may also be used to delete queues which have not been referenced within a given interval.

Syntax

CEMT SET TSQUEUE



Options

(*value*)
Specifies the identifier of the TS queue.

ALL
Specifies that any changes you request are made to all resources of the specified type that you are authorized to access.

ACTION(*value*)
Specifies the action to be taken on the queue. The values are:

DELETE
The queue is to be deleted.

LASTUSEDINT
If this option is specified, the queue is only deleted if its last used interval is greater or equal to the value specified.

POOLNAME(*data-value***)**
Specifies an 8-character pool name.

SYSID(*data-value***)**
Specifies a 4-character shared sysid.

TSQUEUE(*data-value***)**
Specifies the 16-character identifier of the TS queue.

CEMT SET UOW

Commit or back out a shunted unit of work.

Description

The SET UOW command enables you to commit, back out, or force a unit of work that has been shunted during the transaction's indoubt period.

Syntax

CEMT SET UOW

Options

(*value***)**
specifies the 16-character identifier of the UOW to be committed, backed out, or forced.

ALL
specifies that any changes you request are made to all resources of the specified type that you are authorized to access.

BACKOUT
Attempt to force syncpoint backout processing, as specified for this UOW.

COMMIT
Attempt to force syncpoint commit processing, as specified for this UOW.

FORCE
Attempt to force the UOW to backout or commit, as specified on the ACTION option of the TRANSACTION resource definition.

Note that all these values are valid only for UOWs that have been shunted indoubt. For information about the indoubt attributes of TRANSACTION definitions, see the *CICS Resource Definition Guide*.

CEMT SET UOWLINK

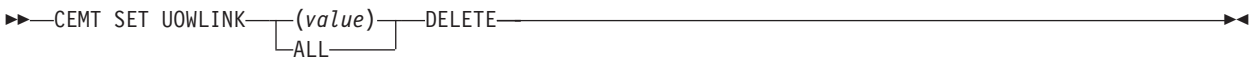
Delete a link to a unit of work that was created by a connection that has since been discarded. UOWLINKS associated with RRS can be deleted when RRS has cold started.

Description

The association between a unit of work and a connection is known as a UOW-link. You can use the INQUIRE UOWLINK command to browse all the UOW-links currently in the system. Some of the UOW-links may have been created by connections that have since been discarded. If so, you may be able to use the SET UOWLINK command to delete them. (For information about when it is safe to delete UOW-links, see the *CICS Intercommunication Guide*..)

Syntax

CEMT SET UOWLINK



Options

(*value***)**
specifies the 4-character identifier of the UOW-connection dependency (the UOW-link) to which this command applies.

ALL
specifies that any changes you request are made to all resources of the specified type that you are authorized to access.

DELETE
specifies that the UOW-link is to be deleted.

CEMT SET URIMAP

Enable or disable a URIMAP definition, and apply or remove redirection for a URIMAP definition.

In the CICS Explorer, the URI maps operations view provides a functional equivalent to this command.

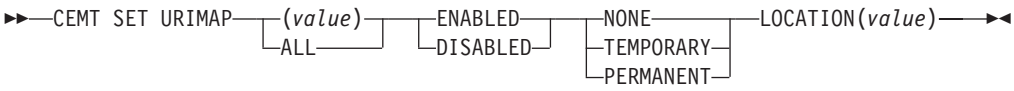
Description

If you disable a URIMAP definition, CICS returns an HTTP response to the Web client with a 503 (Service Unavailable) status code.

You can use the redirection options to set redirection for matching HTTP requests, and specify a URL to which the requests are redirected. You can use this command to apply redirection to an existing URIMAP definition, for example if the application that would normally respond to the HTTP request is unavailable. You can also use this command to remove redirection from a URIMAP definition.

Syntax

CEMT SET URIMAP



Options

(*value***)**
specifies the 8-character name of the URIMAP resource definition to which this command applies.

ENABLED
The URIMAP definition can be accessed.

DISABLED
The URIMAP definition cannot be accessed.

NONE
Requests are not redirected. Any URL specified by the LOCATION option is ignored.

TEMPORARY
Requests are redirected on a temporary basis. The status code used for the response is 302 (Found).

PERMANENT
Requests are redirected permanently. The status code used for the response is 301 (Moved Permanently).

LOCATION *(value)*
specifies a URL of up to 255 characters, to which matching HTTP requests from Web clients can be redirected. This must be a complete URL, including scheme, host, and path components, and appropriate delimiters. CICS checks that the URL is complete and correctly delimited, but CICS does not check that the destination is valid.

The REDIRECTTYPE option is used to specify the type of redirection. If temporary or permanent redirection is specified, the URL in the LOCATION attribute is used for redirection. If no redirection is specified, the URL in the LOCATION option is ignored.

CEMT SET VTAM

Open or close the z/OS Communications Server ACB, set the persistent session delay interval, or deregister CICS from membership of a generic resource.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Syntax

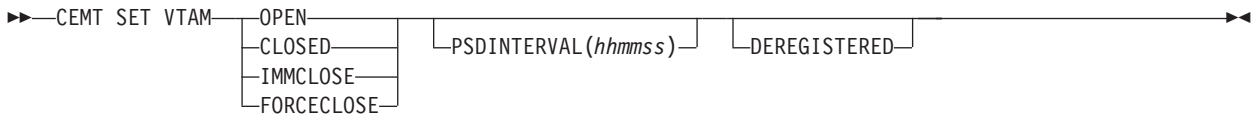
Note: VTAM is now the z/OS Communications Server.

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type CEMT SET VTAM (or suitable abbreviations for the keywords). The resulting display lists the current status, similar to that obtained by using the **CEMT INQUIRE VTAM** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type CEMT SET VTAM (or suitable abbreviations for the keywords), followed by one or more attribute settings that you want to change. For example, cemt s vt o specifies that the connection between CICS and the Communications Server is established.

Type ? at the beginning of either the first or second line to give a syntax prompt. Resetting the values takes effect immediately.

CEMT SET VTAM



Options

CLOSED
The connection between CICS and the Communications Server is to end normally; that is, transactions are allowed to complete and terminals are released normally.

DEREGISTERED
Specifies that CICS is to be removed from the Communications Server generic resource of which it is currently a member. If you deregister a region from membership of a generic resource, you must end any affinities that it owns. See the PERFORM ENDAFFINITY command.

Generic resources are described in the *CICS Intercommunication Guide*.

FORCECLOSE
The connection is stopped immediately, and all terminal connections using the Communications Server ACB (access method control block), with the transactions associated with them, are stopped abnormally. If transactions do not stop because of their state at the time FORCECLOSE was issued, you must issue CEMT SET TASK(nn) PURGE or FORCEPURGE before the ACB can be reopened.

The difference between IMMCLOSE and FORCECLOSE is that IMMCLOSE cancels any task at a Communications Server SNA logical unit (LU) and then closes the LU sessions tidily. FORCECLOSE cancels any task and closes the Communications Server ACB, closing all sessions immediately.

IMMCLOSE
The connection between CICS and the Communications Server is to be stopped immediately. Any transactions using Communications Server-connected LUs are stopped abnormally, and all such LUs are lost to CICS.

The difference between IMMCLOSE and FORCECLOSE is that IMMCLOSE cancels any task at a Communications Server SNA LU and then closes the LU sessions tidily. FORCECLOSE cancels any task and closes the Communications Server ACB, stopping all sessions immediately.

OPEN
The connection between CICS and the Communications Server is established. With ACF/Communications Server, this event occurs at the opening of the ACF/Communications Server ACB through which ACF/Communications Server communicates with CICS.

A **CEMT SET VTAM OPEN** command fails with a STILL CLOSING message if the command is issued while a previous predatory takeover is still being processed. No attempt is made to open the Communications Server ACB, because it is still being closed because of the predatory takeover. After the predatory takeover completes, the **CEMT SET VTAM OPEN** command works as usual. The failure is notified by a RESP=INVREQ and RESP2=14 condition.

If CICS is using Communications Server multinode persistent sessions, and the Communications Server has been restarted after an abend, opening the Communications Server ACB causes CICS to restore the persistent sessions that the Communications Server has retained. However, CICS does not restore APPC synclevel 2 sessions, which are unbound.

PSDINTERVAL (*hhmmss*)
Specifies the persistent session delay interval, expressed in the form *hhmmss*. It specifies if and for how long Communications Server sessions are held in recovery pending state in the event of a failure. The permitted range is 000000 to 235959 inclusive.

000000 If a failure occurs, sessions are ended. This value is the default, and means that persistent sessions support is not exploited.

HHMMSS
Sessions are held in recovery pending state by the Communications Server for up to the interval specified.

Table 8 shows possible CEMT error messages resulting from unsuccessful attempts to set the persistent session delay interval. The listing of RESP2 values for the EXEC CICS SET VTAM command explains the meaning of these messages in more detail.

Table 8. CEMT error messages for attempts to set the persistent session delay interval

Message	RESP/RESP2
INVALID PSDINT	RESP=INVREQ, RESP2=4
NOT WITH XRF	RESP=INVREQ, RESP2=8
SETLOGON FAILURE	RESP=INVREQ, RESP2=9
BACK LEVEL VTAM	RESP=INVREQ, RESP2=10
ACB CLOSED	RESP=INVREQ, RESP2=12
NOT WITH PSTYPE=NOPS	RESP=INVREQ, RESP2=22
RECOVERY ERROR	All other cases of INVREQ

CEMT SET WEB

Change web garbage collection settings, or web 3270 terminal timeout settings.

In the CICS Explorer, the Regions operations view provides a functional equivalent to this command.

Input

Press the Clear key to clear the screen. There are two ways to start this transaction:

- Type `CEMT SET WEB` (or suitable abbreviations for the keywords). The resulting display lists the current status, similar to that obtained by using the **CEMT INQUIRE WEB** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET WEB` (or suitable abbreviations for the keywords), followed by one or more attribute settings that you want to change. For example, `cemt s we g` specifies that you want to change the GARBAGEINT option.

Typing ? at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET WEB

Displayed fields

GARBAGEINT
The interval at which the web garbage collection task runs to clean up web 3270 state data for which the terminal timeout interval has expired. The interval is measured in minutes.

TIMEOUTINT
The period of time after which inactive web 3270 sessions are eligible for garbage collection. The period is measured in minutes.

CEMT SET WEBSERVICE

Set the validation status of a WEBSERVICE resource.

In the CICS Explorer, the Web services operations view provides a functional equivalent to this command.

Description

If the optional attributes are omitted, the command has no effect.

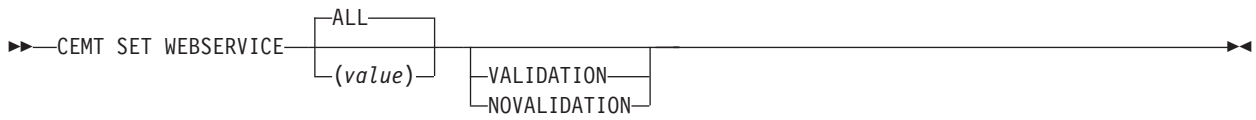
Syntax

Press the Clear key to clear the screen. You can start this transaction in two ways:

- Type `CEMT SET WEBSERVICE` (or suitable abbreviations for the keywords) followed by one or more WEBSERVICE identifiers, or ALL. The resulting display lists the current status, similar to that obtained by using the **CEMT INQUIRE WEBSERVICE** command. You can tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET WEBSERVICE` (or suitable abbreviations for the keywords), followed by one or more WEBSERVICE identifiers or ALL, followed in turn by one or more attribute settings that you want to change. For example, `CEMT SET WEBSERVICE AL V` resets the validation status for all WEBSERVICE resources.

Typing ? at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

SET WEBSERVICE



Conditions: NOTAUTH, NOTFND

Options

ALL
Any changes you request are made to all resources of the specified type that you are authorized to access.

(*value*)
Specifies the name of the WEBSERVICE resource. If the resource is dynamically installed using a pipeline scan, the name can be up to 32 characters in length. Otherwise, the name can be 1 - 8 characters in length.

VALIDATIONST
Specifies whether full validation of SOAP messages is required for this web service:

VALIDATION
Full validation is required.

NOVALIDATION
Full validation is not required.

The WEBSERVICE must be INSERVICE before you can turn validation on and off.

CEMT SET WORKREQUEST

Function

Purge or force purge a specific work request task.

Description

Use the **SET WORKREQUEST** command to terminate specific work requests. You cannot specify a list of identifiers, nor can you use the symbols * and + to specify a family of work requests. If you do not set a work type for the work requests, either IIOP or SOAP, CICS uses IIOP as the default value.

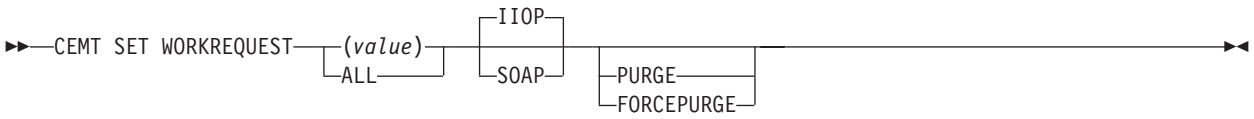
Syntax

Press the Clear key to clear the screen. There are two ways of starting this transaction:

- Type CEMT SET WORKREQUEST with (*value*) or ALL. You get a display that lists the current status, similar to that obtained by **CEMT INQUIRE WORKREQUEST**. You can then tab to the Purgetype field and enter or overtype it with the required value (PUrge or FOrcepurge).
- Type CEMT SET WORKREQUEST with (*value*) or ALL, followed by the attribute setting that you want to change.

Typing ? at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

CEMT SET WORKREQUEST



Options

(*value*)
An 8-character string to identify the work request. The valid characters for this token are the 16 hexadecimal digits, 0–9 and A–F. The token is generated by CICS. Normal usage begins with an INQUIRE WORKREQUEST BROWSE command, to identify the token which is of interest. Then the specific token can be used to issue INQUIRE and SET commands that relate to the task in question.

ALL
Any action you request is performed for all work requests of the specified type that you are authorized to access.

FORCEPURGE
The work request is terminated immediately. System integrity is not guaranteed. In some extreme cases, for example if a work request is forcepurged during backout processing, CICS terminates abnormally. If you want to terminate a work request but do not want to terminate CICS, you should use PURGE instead of FORCEPURGE.

IIOP
Specifies that any action you request is limited to IIOP work requests. This value is the default.

SOAP
Specifies that any action you request is limited to Web service work requests.

PURGE
The work request is terminated, but termination occurs only when system and data integrity can be maintained.

CEMT SET XMLTRANSFORM

Change the validation status of an XMLTRANSFORM resource.

In the CICS Explorer, the XML transforms operations view provides a functional equivalent to this command.

If the optional attributes are omitted, the command has no effect.

Syntax

Press the Clear key to clear the screen. There are two ways of starting this transaction:

- Type `CEMT SET XMLTRANSFORM` followed by one or more identifiers or ALL. You get a display that lists the current status, similar to that obtained by **`CEMT INQUIRE XMLTRANSFORM`**. You can then tab to the highlighted or blank fields and overtype them with the required values.
- Type `CEMT SET XMLTRANSFORM` followed by one or more identifiers or ALL, followed in turn by one or more attribute settings that you want to change. For example, `CEMT SET XMLTRANSFORM AL V` resets the validation status for all XMLTRANSFORM resources.

Typing ? at the beginning of either the first or second line gives a syntax prompt. Resetting the values takes effect immediately.

SET XMLTRANSFORM

Conditions: NOTAUTH, NOTFND

Options

ALL
Any changes you request are made to all resources of the specified type that you are authorized to access.

(*value*)
specifies the 1–32 character name of the XMLTRANSFORM resource.

VALIDATIONST
Specifies whether full validation is required for this XMLTRANSFORM resource:

VALIDATION
Full validation is required.

NOVALIDATION
Full validation is not required.

The XMLTRANSFORM resource must be enabled before you can turn validation on and off.

Chapter 16. CEOT - terminal status

The CEOT command allows you to inquire about the status of your own terminal, and to change some of its values. You cannot use CEOT to inquire about or change the values of any other terminal.

A number of attributes on CSD resource definitions, particularly those that refer to Java applications, are specified using a mixed-case character set. The CEOT upper case translate status options (see "CEOT in set mode" on page 534) enable you to switch between the uppercase translation options to suit a specific requirement. For example, you might want to suppress CICS uppercase translation before using the CEDA transaction to define resources with attributes that require mixed-case input, such as HFSFILE, SHELF, CERTIFICATE, or DESCRIPTION.

You can start CEOT from 3270 family display devices, or from equivalent terminals, or from the operating system console.

Press the Clear key to clear the screen. There are two ways of using this transaction: in inquiry mode and in set mode.

Using CEOT in inquiry mode

About this task

Type `CEOT`. You get the following display:

```
 STATUS:  RESULTS - OVERTYPE TO MODIFY
  Ter(TC37) Trans(CEOT) Pri(000) Pag Ins Ati Tti
     Net(IYCQTC37) Acq Tra

  CEOT SYNTAX:
   < PAgeable | AUtopageable >
   < ATi | NOAti >
   < TTi | NOTti >
   < Uctran | NOUctran | TRANIdonly >









                                          SYSID=HT61 APPLID=CICSHT61
   RESPONSE: NORMAL                    TIME:  13.47.07  DATE: 02.01.01
PF 1 HELP       3 END       5 VAR             9 MSG
```

Figure 126. CEOT transaction: initial screen

The top half of the screen shows the current status of your terminal. The bottom half of the screen shows the syntax of the CEOT command.

Note: NET, ACQ, and CRE are not displayed for non-z/OS Communications Server terminals.

© Copyright IBM Corp. 1982, 2011

The status items are displayed in the following sequence.

Note: If an item has a "negative" value—for example, NOATI—that value will be replaced by a blank space.

Acq
CICS is in session with the logical unit represented by this terminal. You cannot overtype this value.

Aut
See 'AUTOPAGEABLE' in the Command Options below.

The paging status can be modified by overtyping.

ATi
See 'ATI' in the Command Options below.

Cre
This terminal can be acquired automatically by ATI transactions. You cannot overtype this value.

Ins
Your terminal is INSERVICE. You cannot overtype this value.

Net (*netname***)**
displays the name by which this terminal is known to z/OS Communications Server.

Pag
See 'PAGEABLE' in the Command Options below.

The paging status can be modified by overtyping.

Pri (*nnn***)**
displays the priority of your terminal relative to other terminals, and is a number in the range 0–255.

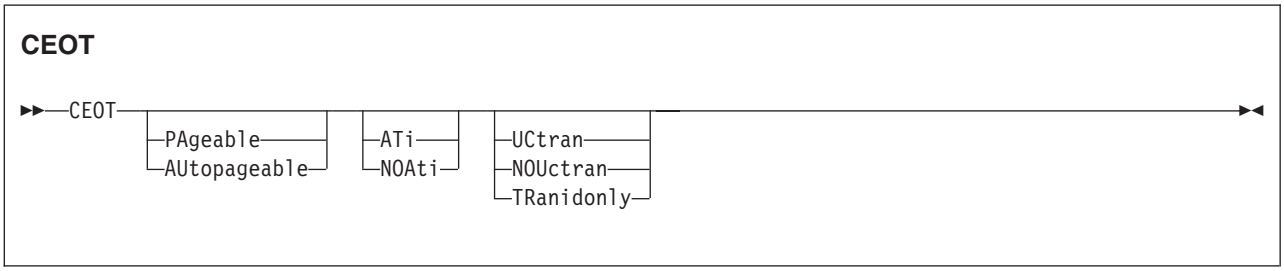
Ter (*xxxx***)**
displays the 4-character identifier of your terminal.

Tra (*ceot***)**
displays the transaction that is running on your terminal. This is always CEOT.

Tti
The terminal is available for use by transactions initiated from this terminal. You cannot change the TTI setting by overtyping it with blanks.

CEOT in set mode

Type `CEOT` followed by one or more of the following attributes:



ATi
Your terminal is available for use by transactions that are automatically

initiated from within CICS or, if the terminal is an ISC session, by transactions that are using this session as an alternate facility to communicate with another system.

AUtopageable
Pages after the first in a series are written to the terminal automatically. This option should never be used with a display device.

NOAti
Your terminal is not available for use by transactions that are automatically initiated from within CICS or, if the terminal is an ISC session, by transactions that are using this session as an alternate facility to communicate with another system.

NOTti
This terminal cannot be used by transactions that are initiated from this terminal. Setting NOTTI is valid only for sequential devices.

NOUCTRAN
The uppercase translation status of your terminal is set to OFF for the current session.

PAgeable
Pages after the first in a series are written to the terminal on request from the operator.

TRANIDONLY
The uppercase translation status of your terminal is set to translate only transaction identifiers entered at the terminal for the current session.

TTi
This terminal can be used by transactions.

UCTRAN
The uppercase translation status of your terminal is set to ON for the current session.

Typing ? at the beginning of the first line gives a syntax prompt. Resetting the values takes effect immediately.

If you do not want to view the current status of your terminal (as displayed in the top half of the CEOT screen), you can change the paging status and the ATI status by using the CEOT syntax displayed in the bottom half of the screen. So, for example, if you want to make your terminal PAGEABLE, type `CEOT PAGEABLE` on the command line of your display and press the Enter key.

Note:

1. You can use CEOT with the routing transaction (CRTE).
   - A user can inquire only about his or her own terminal, and cannot change any of the attributes.
   - The attributes returned are those of the remote terminal TCTTE in the application-owning region(AOR).
   - You must cancel the CRTE transaction by typing CANCEL on the command line.
2. Using CEOT in the TOR:
   - When a terminal is defined as unshippable, and CEOT is used to change the status of the terminal, the changes are not reflected in a remote terminal built

in an AOR as a result of a transaction routing request. The status of the remote terminal is determined by the remote definition of the terminal in the AOR.

- Even when a terminal is defined as shippable, a status change made in the TOR, using CEOT, is not reflected in an existing remote terminal definition in the AOR. A remote terminal in the AOR is created from a shippable definition the first time it is required in the AOR for running a program. The status of the remote terminal returned is its current status in the TOR. Any later updates are not passed on to the remote terminal.

3. If you make an invalid request, CICS lists the keyword options and waits for a valid keyword entry.

Chapter 17. CESF - sign off

Use the CESF transaction to sign off from the CICS system.

This transaction:

- Writes the signoff message to your terminal
- Logs the message on the transient data destination CSCS
- Logs any error message on the transient data destination CSMT.

If you are using CESF to disconnect a terminal and security is active, a message will still be sent to the log if the signoff fails. However, this will not stop the terminal being disconnected if it is a preset terminal or if the terminal is not signed on.

You cannot use CESF on:

- Fixed-format terminals (such as the 3608, the 3614, and the 3653)
- Numeric-only terminals.

For these devices, you must code your own transaction using the EXEC CICS SIGNOFF command. For programming information about this command, see the *CICS Application Programming Reference*.

The syntax of the CESF transaction is as follows:

If you type CESF by itself, you are signed off the CICS system. Your terminal can still start transactions without you having to request them, and messages can be sent to it. You can sign on again at that terminal at any time.

The effects of typing CESF with the GOODNIGHT value or the LOGOFF value depend on the access method in use, as shown in Table 9:

Table 9. Effect of CESF LOGOFF and GOODNIGHT with different access methods

	VTAM DISCREQ=YES	**Otherwise**
CESF	TTI	TTI
CESF LOGOFF	Disc, TTI	TTI
CESF GOODNIGHT	Disc, TTI	NOTTI

Note: VTAM is now z/OS Communications Server.

where:

Disc The terminal is disconnected.

© Copyright IBM Corp. 1982, 2011

TTI Terminal transaction initiation is allowed.

NOTTI
Terminal transaction initiation is not allowed, and your terminal is placed in RECEIVE status so that, the next time a connection is made, it can receive messages.

DISCREQ=YES is an operand of the TYPETERM definition associated with the terminal, and means that CICS is to honor a disconnect request. If DISCREQ=NO, a disconnect request (for example, a CESF LOGOFF request) is not honored.

If a CESF GOODNIGHT is given when there are outstanding automatic transaction initiation (ATI) requests, the ATI requests are actioned and the session is reacquired irrespective of the RDO options.

Chapter 18. CESL - sign-on long

Use CESL to sign on to CICS using a password or a password phrase as authorization.

With CESL, you can sign on to CICS with a password phrase of 9- to 100-characters and a standard password of up to 8 characters. If you enter a password that is 9- to 100-characters, CESL treats the password as a password phrase. In other respects CESL operates in the same way as the CESN sign-on transaction. CESN does not support password phrases but you can specify CESN as an alias for CESL. See "Changing CESN to permit password phrases" on page 548 for more information about how to create an alias for CESL.

When you log on to CICS as a z/OS Communications Server application, by default, you see the CICS "good-morning" panel displayed as shown in Figure 127. This panel is displayed by the CICS-supplied transaction, CSGM. Instead of using CSGM, you can specify your own panel by specifying a different transaction on the GMTRAN system initialization parameter.

```
DFHZC2312 ***  WELCOME TO CICS  *** 12:56:28




                 ******\  ******\  ******\   ******\(R)
                ********\ ******\ ********\ ********\
                **\\\\**\   **\\\ **\\\\**\ **\\\\**\
                **\    \\   **\   **\    \\ **\    \\
                **\         **\   **\       ********\
                **\         **\   **\        ********\
                **\         **\   **\         \\\\**\
                **\   **\   **\   **\   **\ **\   **\
                ********\ ******\ ********\ ********\
                 ******\\ ******\  ******\\  ******\\
                  \\\\\\   \\\\\\   \\\\\\    \\\\\\
```

Figure 127. The CICS good-morning panel

Press CLEAR and type `CESL` on the command line as the CICS sign-on transaction ID. CICS displays the sign-on panel shown in "CESL: IBM 3270 family display devices" on page 540.

The user ID specified for this transaction is used in messages to identify the user.

Notes:

1. If you sign on at a terminal that is already signed on by completing the CESL panel before pressing ENTER, CESL signs off the previous operator. For more information about this, see Using CESL when already signed-on in CICS Supplied Transactions.
2. For numeric-only devices or for the IBM 2741 Communication Terminal, use the signon procedure described in CESL: Non-3270 family terminal in CICS Supplied Transactions.
3. Password phrases and standard passwords are always mixed case but user IDs specified in the CESL transaction are translated to uppercase, ignoring the

© Copyright IBM Corp. 1982, 2011

setting of the UCTRAN attribute. However, if support for mixed case passwords is active in the external security manager, CICS saves the value of the UCTRAN option and temporarily sets UCTRAN to UCTRAN(NO) while you are prompted to enter a user ID and password. CESL then converts the user ID to uppercase but leaves the password in the case entered. When the CESL transaction completes, the UCTRAN attribute is restored to its original value. However, if you disconnect from the terminal before the CESL transaction has completed, the terminal might be left with the UCTRAN(NO) attribute set.

4. You cannot use your standard password as authorization to enter a new password phrase, neither can you use your password phrase to authorize a new standard password.
5. Autoinstall of BMS maps is not carried out in the CESL transaction. When CESL tries to send out any BMS map that is not installed, CESL sends out the English BMS map.
6. The CESL dialog will not prompt you for a new password or password phrase unless your old one has expired but you can use CESL to change your password or password phrase at any time.

You can customize the CESL map set but only if you do not change the format of the symbolic map set. The source for the CESL map is included in map set DFHSNPE in the SDFHSAMP library. After customization, regenerate the physical map sets in the usual way into a load library, which must be included in your DFHRPL concatenation in front of the CICS libraries. See Basic mapping support in CICS Application Programming for more information.

CESL: IBM 3270 family display devices

You can sign on to CICS from most 3270 display devices with the CESL transaction using either your password or password phrase as authorization.

Clear the screen, and type `CESL` on the command line of the screen. Press ENTER. The panel shown in Figure 128 on page 541 is displayed.

```
         Sign on for CICS Transaction Server                APPLID=CICSA100

. . . . . . This is where the good morning message appears. . . . . .
. . . . . . It can be up to four lines in depth to contain the. . . . .
. . . . . . maximum message length of 246 characters (that is, three. .
. . . . . . full lines and six characters on the fourth line) . . . . .



 Type your userid and password, then press ENTER:

         Userid . . . . ________    Groupid . . . ________

         Password . . . ________________________________________________
_____________________________________________

         Language . . . ___

      New Password . . . ________________________________________________
_____________________________________________




 DFHCE3520 Please type your userid.
 DFHCE3540 Ensure that passwords are entered in the correct case.
 F3=Exit
```

Figure 128. CESL transaction: initial screen

Figure 129 shows the panel for 40 x 12 size screens. Because of size limitations, this panel does not support password phrases and there is no space for the release number or the good morning message.

```
         Sign on for CICS Transaction Server                APPLID=CICSA100

            Userid   . . . . ________
            Groupid  . . . . ________
            Password . . . . ________
            Language . . . . ___
            New Password . . ________




 DFHCE3520 Please type your userid.
 F3=Exit
```

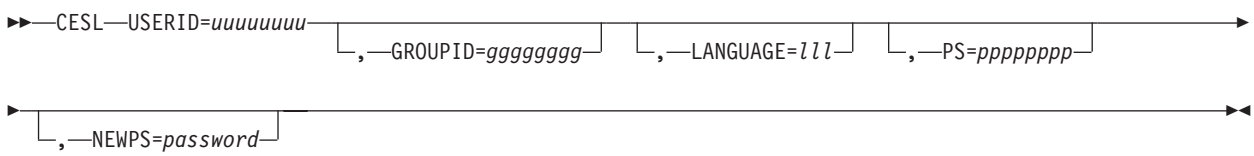
Figure 129. 40 x 12 screen

CESL: Non-3270 family terminals

You can sign on to CICS from non-3270 terminals with the CESL transaction using either your password or password phrase as authorization.

Sign on by entering the following transaction and user information including either your password or password phrase.

CESL



Note: Do not use blanks between parameters in the comma-separated list.

This style of input can be used on video display terminals, but is especially useful for sequential terminals and devices that do not belong to the IBM 3270 family of terminals.

Options

GROUPID=*gggggggg*
your group user ID (up to 8 characters). If this parameter is specified, it might be propagated to remote systems along with the user ID.

LANGUAGE=*lll*
a national language for terminal-directed messages. Enter a national language identifier consisting of a standard, 3-character IBM code from those that are available at your installation. If omitted, CICS uses a national language from one of these sources in order of priority:

1. Language specified for the user in the signon table
2. Language specified in the installed terminal definition for the terminal at which the sign-on occurs
3. Language specified on the NATLANG system initialization parameter.

If you enter a code that is not valid, message DFHCE3537 is issued.

NEWPS=*password*
your new password (up to 8 characters) or password phrase (9 - 100 characters).

- If **PS** is 8 characters or less, it is treated as a standard password and **NEWPS** must also be a standard password. If **PS** is 9 - 100 characters, it is a password phrase, in which case **NEWPS** must also be a password phrase.
- A value that consists entirely of space characters is not interpreted as a password update request. You are not prompted for verification.
- If a password phrase contains blanks or punctuation, the password must be enclosed in single quotes.
- If a password phrase contains single quote characters, each single quote must be enclosed in two more single quotes.
- When a password phrase is entered from a z/OS console the maximum number of characters that you can enter is 120. However, you can increase the number of characters by omitting **NEWPS**. Avoid using many single quotes in your password phrase because this could result in a character string that exceeds this length restriction.

PS=*pppppppp*
your password (up to 8 characters) or password phrase (9- to 100-characters). If the password is required but omitted, RACF reports a security violation before CICS again prompts for the password.

- If a password phrase contains blanks or punctuation, the password must be contained in single quotes.
- If a password phrase contains single quote characters, each single quote must be contained in two more single quotes.
- When entered from a z/OS console the maximum number of characters that you can enter is 120. However, you can increase the number of characters by omitting **PS**.

USERID=*uuuuuuuu*
your user ID (up to 8 characters).

Using CESL when already signed-on

CESL sign-off can occur if you are already signed on before an authorization call to RACF.

When you start the CESL transaction, CICS verifies your user ID and password or password phrase by calling RACF. If the terminal sign-on is valid, the CICS user domain keeps track of your user ID. Thereafter, CICS uses the information about your user ID when calling RACF to make authorization checks. If you were signed on before the call to RACF, you are signed off by CICS whether the returns from RACF are valid or invalid.

If you enter the CESL transaction when you are already signed on with the CESL or CESN transaction, CESL might or might not sign you off. For example, you are not signed off by entering CESL, and then pressing F3 when the sign-on panel is displayed.

In CICS TS for z/OS, Version 4.2, CESL signs off any signed-on user only when a new sign-on attempt is made. A signed-on user is signed off when a user ID is entered on the sign-on panel and the Enter key is pressed. Alternatively, the signed-on user is signed off if the CESL transaction identifier is entered with operands (for example USERID=userid), and the Enter key is pressed.

Unsuccessful sign-on and sign-off errors cause DFHCE*nnnn* messages to be issued.

RACF OIDCARD users can use CESL to sign on if the card reader supports the DFHOPID attention identifier (AID). If it does not, you must write your own sign-on program using the EXEC CICS SIGNON command. For programming information see the *CICS Application Programming Reference*.

Chapter 19. CESN - sign on

Use CESN to sign on to CICS using a password as authorization. CESN does not support password phrases. By default, intending users of CICS see the CICS "good-morning" panel displayed when they log on to CICS as a z/OS Communications Server application

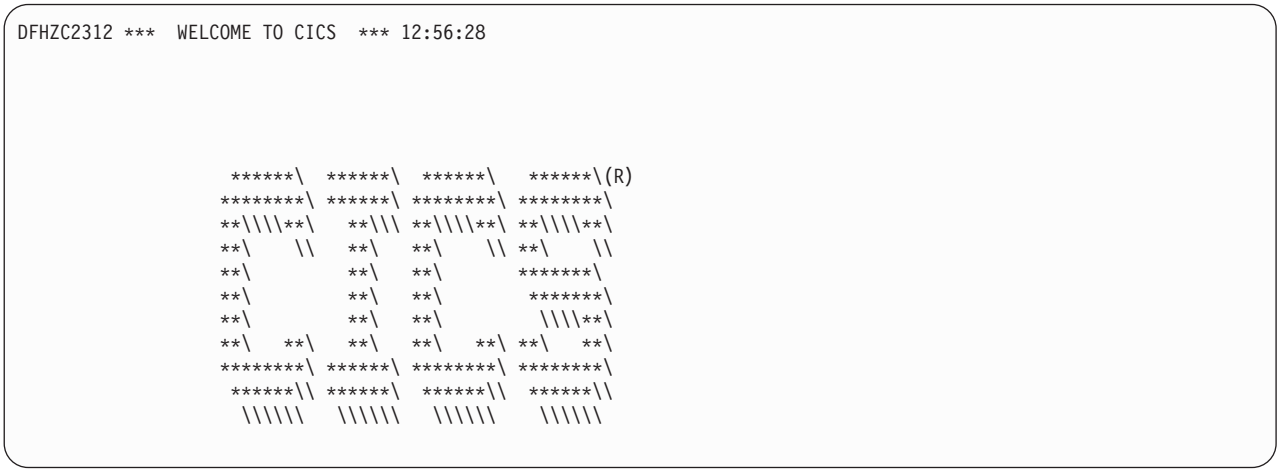


Figure 130. The CICS good-morning panel

This panel is displayed by the CICS-supplied transaction, CSGM. Instead of using CSGM, you can specify your own transaction panel on the GMTRAN system initialization parameter.

Users either press the CLEAR key and type the CICS sign-on transaction ID, CESN or, if CESN is specified as the good-morning transaction, CICS displays the sign-on panel shown in Figure 131 on page 546.

The user ID specified for this transaction is used in error messages and so on, to identify the user.

Note:

1. If you sign on at a terminal that is already signed on, CESN signs off the previous operator. This does not happen, however, unless you complete the CESN panel before pressing ENTER. For further information, see "Using CESN when already signed-on" on page 548.
2. For numeric-only devices or for the IBM 2741 Communication Terminal, use the signon procedure described in "CESN: Non-3270 family terminal" on page 547.
3. User IDs and passwords specified in the CESN transaction are translated to uppercase, ignoring the setting of the UCTRAN attribute. However, if support for mixed case passwords is active in the external security manager, CICS saves the value of the UCTRAN option and temporarily sets it to UCTRAN(NO) while the user is prompted to enter a userid and password. CESN then uppercases the userid but leaves the password in the case entered by the user. When the CESN transaction completes, the UCTRAN attribute is restored to its

© Copyright IBM Corp. 1982, 2011

original value. If the user disconnects from the terminal before the CESN transaction has completed, the terminal might be left with the UCTRAN(NO) attribute set.

4. Autoinstall of BMS maps is not carried out within the CESN transaction. When CESN tries to send out any BMS map that is not installed, CESN will send out the English BMS map.
5. This dialog will not prompt you for a new password unless your old password has expired. If you want to change your password before it has expired, you must use another method, such as signing on at a CICS terminal rather than a console.

You can customize the CESN map set but only if you do not change the format of the symbolic map set. The source for the CESN map is included in map set DFHSNLE in the SDFHSAMP library. After customization, regenerate the physical map sets in the usual way into a load library, which must be included in your DFHRPL concatenation in front of the CICS libraries. See Basic Mapping Support (BMS)

CESN: IBM 3270 family display devices

Clear the screen, and type CESN on the command line of the screen. Press ENTER. You get the following display:

```
         Sign on for CICS Transaction Server                APPLID=CICSA100

. . . . . . This is where the good morning message appears. . . . . .
. . . . . . It can be up to four lines in depth to contain the. . . . .
. . . . . . maximum message length of 246 characters (that is, three. .
. . . . . . full lines and six characters on the fourth line) . . . . .



Type your userid and password, then press ENTER:

         Userid . . . . ________    Groupid . . . ________

         Password . . . ________

         Language . . . __

     New Password . . . ________




DFHCE3520 Please type your userid.
F3=Exit
```

Figure 131. CESN transaction: initial screen

The panel for 40 x 12 size screens is shown below. There is no space for the release number or the good morning message.

```
         Sign on for CICS Transaction Server                APPLID=CICSA100

            Userid   . . . . ________
            Groupid  . . . . ________
            Password . . . . ________
            Language . . . . ___
            New Password . . ________




DFHCE3520 Please type your userid.
F3=Exit
```

Figure 132. 40 x 12 screen

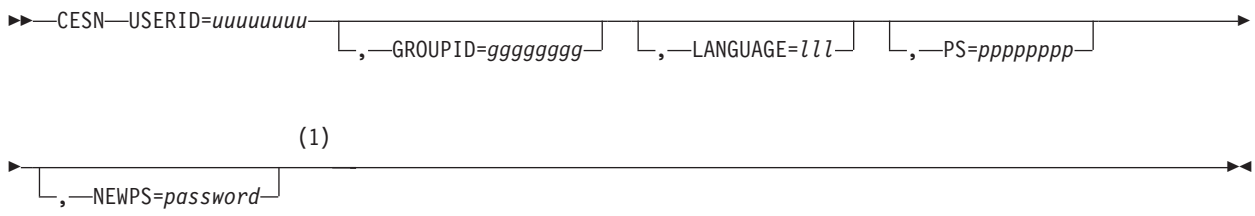
Enter your user ID. This may be up to 8 characters in length.

CESN: Non-3270 family terminal

You can sign on to CICS from non-3270 terminals with the CESN transaction.

Sign on by entering the following transaction and user information. You cannot use a password phrase with CESN. If you want to use your password phrase as authorization, sign on using the CESL transaction.
This style of input can be used on video display terminals, but is especially useful

CESN



Notes:

1 Do not use blanks between parameters in the comma-separated list.

for sequential terminals and devices not belonging to the IBM 3270 family of terminals.

Options

GROUPID=*gggggggg*
your group user ID (up to 8 characters). If this parameter is specified, it might be propagated to remote systems with the user ID.

LANGUAGE=*lll*
a national language for terminal directed messages. Enter a national language identifier consisting of a standard, three-character IBM code from those that are available at your installation. If omitted, CICS uses a national language from one of these sources in order of priority.

1. Language specified for the user in the signon table

2. Language specified in the installed terminal definition for the terminal at which the signon occurs
3. Language specified on the NATLANG system initialization parameter.

If you enter a code that is not valid, message DFHCE3537 is issued:

NEWPS=*password*
your new password (up to 8 characters). A value that consist entirely of space characters is not interpreted as a password update request. You are not prompted for verification.

PS=*pppppppp*
your password (up to 8 characters). If the password is required but omitted, RACF reports a security violation before CICS again prompts for the password.

USERID=*uuuuuuuu*
your user ID (up to 8 characters).

Using CESN when already signed-on

When you start the CESN transaction, CICS verifies your user ID and password by calling RACF. If the terminal sign-on is valid, the CICS user domain keeps track of your user ID. Thereafter, CICS uses the information about your user ID when calling RACF to make authorization checks. If you were signed on before the call to RACF, you are signed off by CICS whether the returns from RACF are valid or invalid.

About this task

If you enter the CESN transaction when you are already signed on with the CESN or CESL, transaction, CESN might or might not sign you off. For example, you are not signed off by entering CESN, and then pressing F3 when the signon panel is displayed.

In CICS TS for z/OS, Version 4.2, CESN signs off any signed-on user only when a new sign-on attempt is made. A signed-on user is signed off when a user ID is entered on the signon panel and the ENTER key is pressed. Alternatively, the signed-on user is signed off if the CESN transaction identifier is entered with operands (for example USERID=userid), and the ENTER key is pressed.

Unsuccessful sign-on and sign-off errors cause DFHCE*nnnn* messages to be issued.

RACF OIDCARD users can use CESN to sign on if the card reader supports the DFHOPID attention identifier (AID). If it does not, you will need to write your own sign-on program using the EXEC CICS SIGNON command. For programming information see the *CICS Application Programming Reference*.

Changing CESN to permit password phrases

Normally when you sign on to CICS with the CESN transaction, you are restricted to using a standard password as authorization. Password phrases are not permitted. However by specifying CESN as an alias for CESL you can change the behavior of CESN so that it invokes the password phrase map.

About this task

To configure CESN to permit password phrases you must alter the CESL definition and change CICS startup to use a modified startup group that does not contain the

CESN transaction. This task assumes that you use the default supplied startup CSD group DFHSIGN. If you are using a copy of DFHSIGN, you can skip step 2 and instead modify your copy in steps 3 and 4.

You can carry out this task using CICS Explorer, BAS, CEDA, or CSD commands.

Procedure

1. If you use the default CICS startup list DFHLIST, specify a new startup list and append the contents of DFHLIST to it. If you already use your own startup list, you can skip this step.
2. Copy the default supplied CSD group DFHSIGN to a different group. If a group with the supplied name does not exist, this operation creates a new one.
3. Remove the CESN transaction from the new group.
4. Update the CESL transaction definition to specify CESN as the alias name.
5. Remove DFHSIGN from your startup list and add the new group in its place. If a list with the name you supply does not exist, a new one is created.
6. Perform a cold restart of the CICS region using the specified startup list in place of DFHLIST.

Chapter 20. CEST - supervisory terminal

Use CEST to invoke a subset of the master terminal (CEMT) transaction. The CEST INQ|SET commands allows you to inquire about and alter some of the values of lines, netnames, tasks, and terminals.

CEST can be invoked from 3270 family display devices, or from equivalent terminals, or from the operating system console.

All of the CEST commands require the SUPRID keyword to be specified. SUPRID identifies the supervisor invoking the transaction, and its value is the 1-or 2-character suffix of a terminal list table (TLT). For information about the SUPRID keyword, see .*CICS Resource Definition Guide*

For example, CEST INQ TERMINAL SUPRID(AL) displays the status of all the terminals in the DFHTLTA1 table. Values can be changed by overtyping the display or by using CEST SET commands. Apart from the requirement that SUPRID has to be specified, the syntax of the commands and the meaning of the values is exactly as described in CEMT INQUIRE|SET LINE, NETNAME, TASK, and TERMINAL.

© Copyright IBM Corp. 1982, 2011

Chapter 21. CETR - trace control

Use CETR to control CICS tracing activity.

To start the transaction, type CETR on the command line, then press Enter.

The following screen is displayed, with values from your system in each field. You can overtype any input field with the new values that you require. When you press Enter, CETR issues the necessary commands to set the new values. If there are any errors, press PF9 to display the error messages. If there is only one short error message, it appears near the bottom of this display.

```
CETR                   CICS Trace Control Facility

Type in your choices.

Item                              Choice       Possible choices

Internal Trace Status       ===>  STOPPED      STArted, STOpped
Internal Trace Table Size   ===>  16      K    16K - 1048576K

Auxiliary Trace Status      ===>  PAUSED       STArted, STOpped, Paused
Auxiliary Trace Data Set     ===>  B            A, B
Auxiliary Switch Status     ===>  ALL          NO, NExt, All

GTF Trace Status            ===>  STARTED      STArted, STOpped

Master System Trace Flag    ===>  OFF          ON, OFf
Master User Trace Flag      ===>  OFF          ON, OFf

When finished, press ENTER.



PF1=Help    3=Quit    4=Components    5=Ter/Trn    6=Pooled JVMs    9=Error List
```

Figure 133. CETR transaction: initial screen

Trace field descriptions

In the example screen, the fields are as follows:

- The internal trace status is STOPPED, so regular tracing is not directed explicitly to the internal trace table. However, the internal trace table is used as a buffer for the other trace destinations, so it always contains the most recent trace entry if at least one trace destination is STARTED.

  The internal trace table is also used as a destination for exception trace entries, which are made whenever CICS detects an exception condition. If an exception condition is detected when the options shown in this example are set, you would find the exception trace entry in the internal trace table as well as in the GTF trace data set.
- The internal trace table size is 16 KB, which is the minimum size. If the internal trace status was STARTED, the trace table would wrap when it became full. Use caution if you specify a large size for a trace table in 31-bit storage, because there must be enough MVS storage available in the CICS region for the internal trace table and the CICS DSAs. For more information about storage use for the internal trace table, see "CICS trace: performance considerations" in the *CICS Performance Guide*.

© Copyright IBM Corp. 1982, 2011

- The current auxiliary trace data set is B, meaning that trace entries are written to DFHBUXT if auxiliary tracing is started. As its status is shown to be PAUSED, no tracing is done to that destination. The auxiliary switch status is ALL, so a switch would be made to the other auxiliary trace data set whenever one became full.
- The GTF trace status is shown to be STARTED, which means that CICS trace entries are written to the GTF trace data set defined to MVS. To write trace entries to GTF, GTF tracing must be started under MVS with the TRACE=USR option.

  If the CICS GTF status is started, but GTF tracing has not been started under MVS, trace entries are not written. In this situation, no error condition is reported.
- The master system trace flag is OFF. This means that no standard tracing is done at all, even though standard tracing might be specified for some tasks. However, special task tracing is not affected. The master system trace flag only determines whether standard task tracing is to be done.

CETR also provides the following functions:

- Set the component trace options (press PF4)
- Set the transaction and terminal trace options (press PF5)
- Set the pooled JVMs trace options (press PF6)

For information about the use of the various CETR options as an aid to problem determination, see the *CICS Problem Determination Guide*.

Defining tracing levels by component

The trace levels for a specific CICS component are represented by two values. One value shows the active level of tracing for standard tracing; the other shows the level for special tracing. In CETR, you can set the active level of tracing for standard or special tracing for an individual component or for a group of components.

About this task

For more information about standard and special tracing, including tracing levels, see the *CICS Problem Determination Guide*.

Procedure

1. Enter `CETR` and press PF4 to display the Component Trace Options screen. The following display is shown, with the values of your own system in each field.

```
CETR                  Component Trace Options                  CI41 IYAHZCCV

Over-type where required and press ENTER.                      PAGE 1 OF 4
Component Standard                          Special
--------  --------------------------------  --------------------------------
  AP      1                                 1-2
  BA      1                                 1-2
  BM      1                                 1
  BR      1                                 1-2
  CP      1                                 1-2
  DC      1                                 1
  DD      1                                 1
  DH      1                                 1-2
  DM      1                                 1-2
  DP      1                                 1–2
  DS      1                                 1-2
  DU      1                                 1-2
  EI      1                                 1
  EJ      1                                 1-2
  EM      1                                 1-2
  EP      1                                 1-2
  FC      1                                 1-2

PF:  1=Help   3=Quit   7=Back  8=Forward  9=Messages   ENTER=Change
```

Figure 134. CETR transaction: component trace options screen

2. Use PF7 and PF8 to scroll backward and forward through the list of components. The SJ domain, which controls JVM tracing, is a special case. Trace levels 29–32 for the SJ component correspond to JVM Level 0 trace, JVM Level 1 trace, JVM Level 2 trace and JVM User trace respectively. You cannot use the component trace options screen to control JVM tracing; use the JVM trace options screens instead, as described in "Specifying trace options for pooled JVMs in CETR" on page 557.
3. Specify the level of tracing that you require for a component. You can specify a range of tracing levels, of which levels '1' and '2' are usually the most appropriate, using any of the following methods:
   - Specify the tracing level by over-typing the displayed value; for example, `1-2` or `1,2`.
   - Type `ALL` in upper or lower case under the relevant heading.
   - Use the equal sign = when changing several fields, to indicate that a particular field should be changed to the same value as the one above.

   If you use CETR to turn on tracing for a particular component, CICS does not save modifications to the component trace flags across any kind of CICS restart.
4. Type `OFF` to turn off tracing for a particular component under either the special or standard heading.

Specifying transaction and terminal-related trace options

In CETR you can control tracing for particular CICS tasks by specifying the transaction ID used to start the task, or the ID for the terminal from which the task is to be run. You can set standard or special tracing for the tasks, or suppress tracing for a task. You can also enable z/OS Communications Server exit tracing and ZCP trace for specific terminal IDs or netnames.

Procedure

- To trace a single transaction using special tracing, perform the following steps:
  1. Enter `CETR` and set the master system trace flag to OFF. This setting turns standard tracing off.

2. Press PF4 to select the levels of tracing for the required components under the Special heading.
3. Return to the main CETR display and press PF5. You get the following display, with the values of your own system in each of the fields:

```
CETR                   Transaction and Terminal Trace           CI41 IYAHZCCV

Type in your choices.

Item                               Choice       Possible choices

Transaction ID              ===>               Any valid 4 character ID
Transaction Status          ===>               STandard, SPecial, SUppressed

Terminal ID                 ===>               Any valid Terminal ID
Netname                     ===>               Any valid Netname
Terminal Status             ===>               STandard, SPecial
Terminal VTAM Exit Trace    ===>               ON, OFf
Terminal ZCP Trace          ===>               ON, OFf

VTAM Exit override          ===>  NONE         All, System, None


When finished, press ENTER.




PF1=Help      3=Quit      6=Cancel Exits      9=Error List
```

Figure 135. CETR transaction: transaction and terminal trace screen

Note: VTAM is now z/OS Communications Server.

4. Specify the transaction ID that you want to trace and a transaction status of SPECIAL.

Setting special tracing for a particular transaction means that special flags for each component are used to decide whether a particular trace entry is to be made at any given point in the process of the task.

- To trace the tasks associated with a particular terminal, perform the following steps:
  1. Set the standard component trace flags to level 1 only.
  2. Set the special component trace flags to level 1 and level 2.
  3. Set the trace for the terminal to special.

  In this way, all tasks run from the specified terminal have level 1 and level 2 tracing. All other tasks have level 1 tracing only.
- To set z/OS Communications Server exit tracing in an AOR, enter the applid of the TOR as Netname and enter ON for the Terminal z/OS Communications Server Exit Trace.
  - If you want to trace the bind flows for an autoinstalled terminal in a CICS region where dynamic LU alias is in operation, the NETNAME you specify must be the real network name. If you use the real network name and there is more than one network using that name, the z/OS Communications Server exit tracing is activated for each occurrence of the network name.
  - If you want to trace an autoinstalled terminal when you know the LU alias, specify the LUALIAS name.

Specifying trace options for pooled JVMs in CETR

Although the trace options for pooled JVMs are part of the SJ component, they are controlled by using the pooled JVMs trace options screens, rather than the component trace options screen. Trace levels 29–32 for the SJ component correspond to pooled JVMs Level 0, Level 1, Level 2, and User trace respectively.

About this task

Pooled JVM trace can produce a large amount of output, so activate pooled JVM trace only for special transactions, rather than turning it on globally for all transactions.

Procedure

1. Enter CETR and press PF6 to display the "Pooled JVMs Trace Options" screen. There are two pooled JVM trace options screens: one for pooled JVM Level 0 trace and pooled JVM Level 1 trace, and one for pooled JVM Level 2 trace and pooled JVM User trace. The screens display the trace flags and the trace options strings that are associated with each flag.

```
 CETR                    Pooled JVMs Trace Options               MAIN IYKHB01

 Type in your choices.                                           PAGE 1 OF 2

 Item                              Standard            Special

 Pooled JVMs Level 0 Trace   ===>  OFF  ON, OFf  ===>  OFF  ON, OFf
 Option String:
LEVEL0



 Pooled JVMs Level 1 Trace   ===>  OFF  ON, OFf  ===>  OFF  ON, OFf
 Option String:
LEVEL1



 When finished, press ENTER.




 PF1=Help    3=Quit    6=Reset             8=Forward    9=Error List
```

Figure 136. CETR transaction: Pooled JVMs trace options screen 1 of 2

2. Optional: To set level 2 or User tracing, press PF8 to move to the second Pooled JVM trace options screen.

```
CETR                          Pooled JVMs Trace Options          MAIN IYKHB01

 Type in your choices.                                           PAGE 2 OF 2

 Item                              Standard            Special

 Pooled JVMs Level 2 Trace   ===>  OFF  ON, OFf  ===>  OFF  ON, OFf
 Option String:
LEVEL2



 Pooled JVMs User Trace      ===>  OFF  ON, OFf  ===>  OFF  ON, OFf
 Option String:
NONE



 When finished, press ENTER.




 PF1=Help    3=Quit    6=Reset    7=Back                9=Error List
```

Figure 137. CETR transaction: Pooled JVMs trace options screen 2 of 2

3. Specify what tracing options you require.
   - Use the free form 240-character field to add parameters that include or exclude particular components or trace point types at the selected trace levels. For details, see Defining and activating tracing for pooled JVMs in Java Applications in CICS.
   - Use the standard and special flags to control tracing.
   - If you require complex conditions for tracing, use the **JVMUSERTRACE** system initialization parameter.

   Set any of the trace options to an asterisk (*) to reset the option to its default settings, or press PF6 to reset all of the JVM trace options to their default settings.
4. Use the Transaction and Terminal Trace screen to switch on the standard and special flags for particular transactions.

Results

When you activate JVM trace, the results appear as CICS trace points in the SJ (JVM) domain. Each JVM trace point that is generated appears as an instance of a CICS trace point:

- SJ 4D02 is the trace point used for formatted JVM trace information.
- SJ 4D01 is used for any JVM trace points that cannot be formatted by CICS. If you see this trace point often, check that the trace format file supplied with the IBM 64-bit SDK for z/OS, Java Technology Edition is present in the `/lib/` subdirectory of your SDK installation. For Version 6.0.1 it is called `J9TraceFormat.dat`. CICS requires this file to format the JVM trace points.

If the JVM trace facility fails, CICS issues the trace point SJ 4D00.

What to do next

Further information about JVM trace and about problem determination for JVMs is in the Java Diagnostics Guide.

Chapter 22. CIDP - inactivate debugging profiles

Transaction CIDP is a utility transaction which you can use to inactivate all debugging profiles in your system. For example, you might choose to run the transaction at the end of the normal working day to ensure that no active profiles remain in a system that is unattended.

You can inactivate all the debugging profiles in your system in the following ways:

Run transaction CIDP from a 3270 terminal
There are no input parameters. The transaction issues a terminal message and a console message which tell you how many debugging profiles have been inactivated.

Invoke program DFHDPIN
You can link to program DFHDPIN from another program, or include it in the second phase Program List Table (PLT). There is no input commarea. The program issues a console message which tells you how many debugging profiles have been inactivated.

For other operations on debugging profiles, use the application debugging profile manager.

© Copyright IBM Corp. 1982, 2011

Chapter 23. CIND - indoubt testing tool

Use the CIND transaction to test the effect of indoubt failures on application programs. You can also use this transaction to produce shunted units of work (UOWs) to test programs that use SPI commands to inquire on, and change the characteristics of, shunted UOWs and unshunt UOWs that have been shunted using CIND.

For a description of the concepts involved in the synchronization of distributed applications, including indoubt periods and shunted UOWs, see the *CICS Intercommunication Guide*.

CIND can operate in a stand-alone system; it provides a way of changing the UOW so that it is distributed, and of causing a failure during syncpoint processing in the indoubt window.

It can be used to cause a failure in syncpoint processing as follows:

- CIND changes the UOW to be distributed by adding itself to the UOW as the syncpoint initiator and coordinator. This has the effect of simulating a link to a remote CICS system that has assumed the syncpoint coordinator role and that has sent a syncpoint PREPARE request to CICS. This in turn forces CICS to assume the role of a syncpoint subordinate. Being a syncpoint subordinate means that during the two-phase commit syncpoint protocol, CICS itself becomes indoubt as to the outcome of the UOW. While CICS is indoubt, CIND simulates a link failure to the syncpoint coordinator.
- If the TRANSACTION resource definition specifies WAIT(YES), the failure in the indoubt window causes CICS to shunt the UOW.
- If the TRANSACTION resource definition specifies WAIT(NO), a forced decision is taken to commit or back out the UOW, as defined by the TRANSACTION ACTION option. In this case, the UOW is not shunted, but is forced to back out or commit without consulting the coordinator of the distributed UOW.

You can also use CIND to shunt a UOW that is genuinely distributed across multiple CICS systems. It must be activated on the CICS system where the syncpoint is initiated, so that it can assume the role of syncpoint initiator and coordinator before any other CICS systems try to do so. In particular, CIND should not be activated for CICS mirror transactions unless the mirror can be guaranteed to be the syncpoint initiator (that is, a mirror running a server transaction that has been DPLed to specifying SYNCONRETURN).

When active, CIND forces all transactions defined to be in a new transaction class, DFHTCIND, to fail indoubt when they reach syncpoint.

Note: You cannot use CIND on internal CICS system transactions. If you change a system transaction so that it is defined in transaction class DFHTCIND, CIND detects that it is to be used with a CICS system task, and issues message DFHIN1014 to CSMT. The system task continues to run and CIND does not cause it to fail indoubt when it reaches syncpoint.

The required resource definitions for CIND are provided in group DFHINDT, which is a member of group list DFHLIST.

© Copyright IBM Corp. 1982, 2011

Running the CIND indoubt tool

Run the CIND indoubt tool from a terminal to test the effects of indoubt failures on application programs. You can also use this transaction to produce shunted units of work and change the characteristics of units of work.

About this task

When active, the tool forces all transactions that are defined in the transaction class, DFHTCIND, to fail in doubt when they reach a synchronization point. You cannot use CIND on internal CICS system transactions.

Procedure

1. Change the transaction class of the transactions that you want to go indoubt to DFHTCIND.
2. Enter `CIND ON`. This command activates the tool and enables a task-related user exit for the task start. You get a message confirming that the tool is active.

   For every transaction that starts, the task-related user exit inquires on the transaction class. If the class is DFHTCIND, the task-related user exit issues a call to the CICS recovery manage to add a link to CIND, acting as a CICS coordinator system called DFHINDSP.
3. Run you transactions. If the transactions update recoverable resources and the TRANSACTION resource specifies the attribute WAIT(YES), they fail in doubt. The transaction fails with abend code ASP1 and the unit of work is shunted. You can inquire on the shunted units of work.

   A transaction that updates nonrecoverable resources, or that issues only READ requests, fails with abend code ASPR. The unit of work is not shunted because it is read-only.
4. Enter `CIND OFF`. This command disables the task-related user exit and stops any transactions going indoubt. You get a message confirming that the tool is no longer active.
5. Enter `CIND RESYNC COMMIT` or `CIND RESYNC BACKOUT`. These commands simulate a coordinator CICS system reconnecting to your CICS region. The coordinator is DFHINDSP.

Results

The CICS recovery manager unshunts all units of work that were previously shunted because of the tool. The commit or backout specifies what action the recovery manager does when the unshunt takes place. For each unit of work that is unshunted, message DFHIN1012 is sent to CSMT. This message details the unit of work status as defined by DFHINDSP and the recovery manager.

You can also use these commands to simulate what happens units of work are not synchronized. For example, if a TRANSACTION resource specifies the attributes WAIT(NO) and WAITACTION(COMMIT), the unit of work is not shunted but forced to commit. If you enter a `CIND RESYNC BACKOUT` at this point, the DFHIN1012 message output shows the unit of work to be out of sync, as does message DFHRM0111.

Inquiring on the status of CIND

About this task

Type `CIND INQUIRE`. This returns the current status of CIND with one of the following messages:

```
DFHIN1003 date time applid.  The indoubt tool
is active for DFHTCIND tranclass transactions.

DFHIN1006 date time applid.  The indoubt tool is
not active.
```

Running the indoubt tool using an EXEC CICS LINK command

About this task

Instead of running the indoubt tool using a terminal and the CIND transaction, you can use the EXEC CICS LINK command to link to the indoubt tool from an application passing a COMMAREA. For example:

```
EXEC CICS LINK PROGRAM('DFHINDT') COMMAREA(COMM)
```

where COMM is a data-area that contains the commands to pass to DFHINDT. COMM should contain one of:

```
' ON '
' OFF '
' RESYNC COMMIT '
' RESYNC BACKOUT '
```

Setting a single transaction indoubt

About this task

If you want a single transaction to go indoubt regardless of its transaction class, an application can link to program DFHINDAP. (No commarea is required in this case.) This causes that single transaction to go indoubt without the need to type CIND ON. This single transaction can be resynchronized by typing CIND RESYNC COMMIT, and so on, as described in "Running the CIND indoubt tool" on page 562.

DFHINDAP works by enabling the indoubt task-related user exit if necessary and then issuing an API call to the indoubt task-related user exit to add the dummy coordinator. This has the advantage of allowing a test application to decide it wants to force indoubt dynamically, after it has started. Hence a UOW, other than the first one in the transaction can be forced indoubt.

CIND Messages

The indoubt tool outputs audit trail DFHINxxxx messages to CSMT at the following times:

- When CIND is activated using CIND ON
- When CIND is deactivated using CIND OFF
- When a resync is initiated using CIND RESYNC COMMIT or CIND RESYNC BACKOUT
- For each UOW acted upon, when:
  - CIND adds itself as coordinator to the UOW

- CIND simulates losing the link to the coordinator system and the UOW is shunted
- CIND simulates losing the link to the coordinator, but the UOW is not shunted
- CIND causes resynchronization of a UOW.

Each message shows the transaction ID, the task number, and the UOW identifier.

Chapter 24. CLDM - CICS load module map

Transaction CLDM is used to generate a CICS load module map. The load module map can be used by the z/OS hardware instrumentation service or during problem determination to help identify a program that has been loaded by the CICS loader domain.

The CICS load module map can be generated in different formats using the options shown in following syntax diagram.

pathname
The name of an existing USS file system path that CLDM uses to store a z/OS hardware instrumentation services CICS load module map. The path name must be an absolute pathname; that is, it must start with a forward slash (/).

Because the USS file system path name is usually lowercase when CLDM is invoked on a terminal or console that is defined with MSGCASE=UPPER, CLDM converts the *pathname* value to lowercase. If the path name is not converted to lowercase, it must be enclosed in single quotes. If the path name is mixed case, ensure that the terminal definition is appropriate and enclose the path name in single quotes.

The file name written to the specified USS file system path is generated by CLDM and has the format SYSHIST*yyyymmdd.hhmmss.asidx.jobname*.CICSMAP, where:

- *yyyymmdd* is the year, month, and day that CLDM ran.
- *hhmmss*is the hour, minute, and second that CLDM ran.
- *asidx* is the 4-byte hexadecimal address space ID of the CICS region.
- *jobname* is the CICS job name.

The format of records written to the USS file system is described in the z/OS System Commands manual, in the table 'Interpreting the information in a .MAP file from hardware data collection'. However the following conditions apply:

- CLDM generates data records only with record types of M (Module) and C (CSECT).
- For module (record type M) records, if the module is in an LPA region (that is MLPA, PLPA, or FLPA), the memory area is always set to M.
- The long name (VOLSER and DSN) field is always blank for module records.

Note: It is the CICS region user ID and not the user ID associated with the CLDM transaction that is checked for authority to create the file in the USS file system.

sysoutclass
The sysout class that CLDM must use to produce a loader program map report. CICS must have the CICS spool interface active (SPOOL= YES in the

© Copyright IBM Corp. 1982, 2011

SIT) for the SYSOUT operand to be effective. No other spool-related attributes must be specified. You can invoke CLDM in the following ways:

- Run the transaction from a 3270 terminal. Input is received from the terminal and messages are sent to the terminal.
- CLDM can be started from a CICS API START command. Input is retrieved from the START command and messages are sent to the console.
- Link to DFHLDMAP. The input is obtained from the current channel in a container DFHLDM.INPUT and messages are written to the container DFHLDM.OUTPUT.

Note: This support is intended to map executable code. Literal pools cannot be included in the CSECT mappings for this release.

Chapter 25. CLER - Language Environment runtime options

The CLER transaction is provided by Language Environment and is not shipped as part of the CICS Transaction Server product.

Use the CLER transaction to display the Language Environment runtime options. You can also modify a subset of the options. If you want to print the runtime options, you can write them to the CESE queue.

CLER is a conversational transactional that presents the display panel shown in Figure 138.

```
 CLER                                                          CICS IYCLZCFE

              LANGUAGE ENVIRONMENT REGION LEVEL RUNTIME OPTIONS


TYPE IN YOUR CHOICES.

RUNTIME OPTION     CHOICE      POSSIBLE CHOICES


ALL31          ==>  ON          ON, OFF
CBLPSHPOP      ==>  ON          ON, OFF
CHECK          ==>  ON          ON, OFF
INFOMSGFILTER ==>  OFF         ON, OFF
RPTOPTS        ==>  OFF         ON, OFF
RPTSTG         ==>  OFF         ON, OFF
TERMTHDACT     ==>  DUMP        QUIET,MSG,TRACE,DUMP,UAONLY,UADUMP,UAIMM,UATRACE
TRAP           ==>  OFF         ON, OFF


WHEN FINISHED, PRESS ENTER.
```

Figure 138. CLER run-time options transaction

Note:

1. The CLER transaction uses basic mapping support (BMS) and therefore this function is available only through those terminal devices that are supported by BMS.
2. The transaction, program, and mapset resource definitions that you require to run CLER are in the CEE group. They are added to the CSD automatically during CICS installation, as part of the DFHCOMDS job. They are also supplied in the Language Environment SCEESAMP library.

For information about how to use the CLER transaction, see *z/OS Language Environment Debugging Guide and Run-Time Messages*.

© Copyright IBM Corp. 1982, 2011

Chapter 26. CMAC - messages and codes display

Use CMAC to display information about CICS messages and codes.

To use the CMAC transaction:

- Add the DFHCMAC group on the CICS system definition (CSD) file to your initialization list (DFHLIST)
- Add the data set name of the CICS messages file to the definition of the DFHCMACD file within the DFHCMAC group.

For information about how to carry out these actions, see the *CICS Resource Definition Guide*.

Note:

1. This transaction is not valid for the MVS console.
2. The information is available only in English.

You start the transaction by typing one of the following:

1. **CMAC**
2. **CMAC abcd** where 'abcd' is the abend code.
3. **CMAC xxxx** where 'xxxx' is the message number only.
4. **CMAC yyxxxx** where 'yy' is the component id and 'xxxx' is the message number.
5. **CMAC DFHxxxx** where 'xxxx' is the message number.
6. **CMAC DFHyyxxxx** where 'yy' is the component id and 'xxxx' is the message number.

If you use method 1, you get the display shown in Figure 139 on page 570.

© Copyright IBM Corp. 1982, 2011

```
DFHCMC01             Display On-line Messages and Codes


Type the required message identifier, then press Enter.



     Component ID ===>  (for example, TC for Terminal Control
                                      FC for File Control, etc.)
                        This field is required for Messages in the
                        form DFHxxyyyy, Where xx is the Component ID.

     Message Number =>  (for example, 1060, 5718, or Abend Code
                        such as ASRA, etc.)









F3=Exit to CICS
```

Figure 139. CMAC transaction: initial screen

You tell CICS which message you want by typing the component ID and the message number. If the message does not have a component ID, leave that field blank.

The display then gives information under the headings:

Explanation
An explanation of the cause of the message.

System action
The actions the system is taking.

User response
What actions you can or cannot take.

Destination
Where to see the message. This can be the console, the end-user terminal, or a transient data queue.

Module
The name of the module reporting the error.

For example, to display information about message DFH5130, which does not have a component ID, type:

```
CMAC 5130
```

You get the following display:

```
DFH5130   UNABLE TO LOCATE MODULE DFHCICS.  PRIMARY
          CSD NOT INITIALIZED.
EXPLANATION:  The DFHCICS module is missing from the library.
SYSTEM ACTION:  Processing of the INITIALIZE command is
terminated.
USER RESPONSE:  Ensure that the DFHCICS module is present in
the library.
DESTINATION:  SYSPRINT
MODULE(S):  DFHCSDUP
F3=CANCEL
```

Figure 140. CMAC transaction: 1st example screen

To display the message CE3528, type a component ID of CE and a message number of 3528. You get the following display:

```
DFHCE3528 Signon failed during SECLABEL checking.


EXPLANATION:   The signon request has failed because the
external security manager (ESM) detected a critical error.

SYSTEM ACTION:  The signon transaction terminates.

USER RESPONSE:  Refer to message DFHSN0108 on the CSCS log
for the information and actions necessary to resolve this
problem.

DESTINATION:  Terminal End User

MODULE(S):  DFHSNP




F3=CANCEL
```

Figure 141. CMAC transaction: 2nd example screen

If you use methods 2 through 6 (listed above), the initial screen is bypassed and you move directly to the message information screen.

Chapter 27. CMSG - message switching

Use CMSG to send messages from your terminal to one or more destinations.

The destination may be:

- Another terminal
- Another terminal, but only when a specified operator is signed on at that terminal
- Another terminal, but only when the operator signed on at that terminal is in a specified class
- Another operator (CICS locates the operator, if currently signed on, and sends the message to that operator's terminal)
- All terminals
- All terminals with operators signed on in specified classes
- Any combination of these destinations.

You can specify that the message is to be sent at some future time. If the message cannot be sent, your terminal is notified.

In addition, standard routing lists can be provided using the terminal list table (TLT), which can include terminal and operator identifiers. Multiple routing lists can be specified at the time the message is entered, and individual destinations can be deleted from, or added to, the lists at this time.

CICS message switching is provided by a service program invoked by a user-defined transaction identifier that must be defined to CICS. This transaction identifier can be CMSG or any user-defined 4-character transaction identifier. CMSG is the transaction identifier assumed throughout this book.

You can extend message switching to include predefined messages and even complete message-switching transaction input formats.

The terminals from which message switching is initiated, and all terminals of message destinations, must support BMS.

Routed messages can be directed to a terminal, including the transaction terminal, that supports partitions. However, such messages reset the terminal to “base” state.

Sending a message

To send a message to another terminal or terminals, enter the transaction identifier, CMSG, followed by the applicable options.

About this task

The options for CMSG are shown in the syntax diagram “CMSG command” on page 578.

Full details of the options are given under “CMSG command options” on page 578.

© Copyright IBM Corp. 1982, 2011

After entering CMSG, you must include at least the message text, a destination (ROUTE, or OPCLASS, or both), and the option SEND (which you can abbreviate to S). For example:

```
CMSG 'GOOD MORNING',R=ALL,S
```

sends the message ‘GOOD MORNING' to all terminals.

If all is well, your message is sent, and the response

```
MESSAGE HAS BEEN ROUTED
```

is returned to your terminal.

This response is received for the normal completion of a message-switching transaction. It means that the message has been successfully routed to its destination but does not reflect any delivery status.

You can use backspace characters in your input.

New-line characters (or carriage returns on a hard-copy terminal) that are not part of the message are changed to commas, unless they appear at the beginning or end of the input, in which case they are deleted. In this way, you can use NL characters as option separators, instead of commas. For example, on a display device, you could enter:

```
CMSG R=(T001,T002)#
T=1500#
'THE TIME IS 1500 HOURS'#
S@
```

where the ENTER key.

You can enter the options, except SEND and CANCEL (one of which must be entered last), in any order after CMSG and one or more spaces. You can abbreviate every option, except CANCEL, to a single character, and you can include one or more options in a single input. If you specify two or more options, you must follow each by a comma, except the last, which must be followed by a space or by an end-of-data (EOD) condition indicating the end of your input. Any data following a space in the same input is ignored.

To mark the end of a single input:

- Press the ENTER key on display devices such as the IBM 3270
- Press carriage return for hardcopy terminals such as the IBM 2741.

To maintain the conversation with a terminal, the status of all inputs, except the final input which includes the SEND option, is saved in a single temporary-storage record.

A message-switching transaction remains conversational until ended. This means that all input (even other transaction identifiers) is passed to the message-switching program. You must end a message-switching transaction (possibly involving multiple inputs) before you can start a different transaction.

Canceling a message

About this task

You can cancel your current message-switching transaction by entering:

- CANCEL as the last 6 characters of your input. You get the response:

  ```
  TERMINATED BY OPERATOR.
  ```

- CMSG as your first 4 characters, followed by end-of-data (EOD). You get the response:

  ```
  NO INPUT - REENTER.
  ```

- CMSGx as your first 5 characters, where x is any character other than a space or a C. You get the response:

  ```
  SPACE MUST FOLLOW TRANSID.
  ```

- CMSG as your first 4 characters followed by a space and EOD. This starts a new message-switching transaction. You get the response:

  ```
  CONTINUE INPUT.
  ```

The first three examples in this list terminate the conversation and allow you to enter other transactions.

Message-switching continuation

The message-switching transaction appears to be conversational. Internally, a new task is created to process each message switching input.

If a task requires more information to complete the transaction, the current status of the transaction is saved in temporary storage, a response is sent to your terminal, and the task terminates with a requirement that the next task started be a message-switching task.

The saving of status replaces any previously saved status for any message-switching transaction from that terminal.

If a message-switching transaction completes successfully, or you have canceled it, an appropriate response is sent to your terminal, the status of the transaction is not saved, and the task terminates with no transaction restart specified. The saved status is always used by a message-switching task to continue a conversational input from your terminal. You can recall the saved status by entering the transaction identifier to which you have added C (for example CMSGC). You can include additional input, but it must follow CMSGC with no intervening spaces.

This method is useful because you can enter CANCEL to bypass errors in the current input and yet not allow the conversation to continue. You can then enter, for example CMSGC and then enter EOD, which puts the terminal in the conversation that prevailed at the time of the most recent input that has not been canceled or has not resulted in a message being routed. At this point you could also add an option if desired, for example CMSGCT=1030. (Note that there are no spaces between the transaction identifier, the C, or the following data.)

For example, enter:

```
CMSG R=(T40,T41)
```

and you receive:

```
R OK CONTINUE INPUT,
T=1500
RT OK CONTINUE INPUT
'THERE WILL BE ..... AT THE
RT OK CONTINUE MSG
MAIN NIO;FOMH PM VS;OGPTMOS SBR, CANCEL (see Note)
TERMINATED BY OPERATOR
CMSGCMAIN BUILDING ON CALIFORNIA AVENUE
RT OK CONTINUE MSG
IN ROOM 407',SEND
MRTS OK MESSAGE HAS BEEN ROUTED
```

Note: You can cancel and reenter instead of trying to correct the input. This terminates the entire transaction. To save the previously entered input, enter CMSGC to restart at the previously saved status point and continue the message.

Your terminal is free between a message-switching response and the next input, and if it is in TRANSCEIVE status, an automatically initiated task can be started on your terminal before you can continue the message-switching transaction.

Upon completion of this new task, even if it required a response from you, the message-switching transaction continues as if nothing had intervened. However, if you receive a response indicating that the intervening task has issued a RETURN, you must enter the transaction identifier followed by C to force continuation of the previous message-switching transaction.

Message delivery

You can deliver a message to a destination if the following conditions are met:

- The specified delivery time has been reached
- The terminal is INSERVICE, and either:
  - The opid of the operator signed-on matches the requested opid if specified in the ROUTE option

    or
  - The class of the operator signed-on matches any class specified by the OPCLASS option.

The processing and paging status of a terminal, which also affects delivery of eligible messages, is initially specified on the CEDA DEFINE TYPETERM command. You can change the status using the CEMT, CEST, and CEOT transactions.

When the processing status is RECEIVE or TRANSCEIVE, a message is sent automatically to the terminal when no transaction is attached to it.

When the processing status is TRANSACTION, you must request message delivery by entering a paging command, see Chapter 32, “CSPG - page retrieval,” on page 619.

A message is formed into one or more “pages”, the size of which is predefined for a destination by the system programmer. The terminal processing status determines the delivery of the first or only page of a message. Succeeding pages, if any, are sent according to the paging status of the terminal as defined by the system programmer.

When the terminal is in PAGE status, you must request succeeding pages through a paging command.

When the terminal is in AUTOPAGE status, the complete message is printed.

Subsystems and terminal systems

You can use the CMSG message-switching facility ROUTE option to manage responses from display devices, communication terminals, and sequential terminals. Variable conditions requiring a response cannot always be anticipated, and the message-switching task performs this function by reading input and issuing error messages until a valid response is received.

Except for any specific details relating to the ROUTE option (see CMSG command options, ROUTE), all other information about the use of the following subsystems in message switching is given in the appropriate IBM subsystem guides:

- IBM 3600 Finance Communication System (z/OS Communications Server), SC33-0233
- IBM 3650 Retail Store System, SC33-0234
- IBM 3770 Data Communication System, SC33-0235
- IBM 3790 Communication System, SC33-0236.

IBM 3270 display devices

The responses to 3270 family display devices are designed to fit on a single 40-character line of a Model 1 display. To provide more space for the response, the indicators that precede the response do not have the spaces between them that appear on non-3270 terminals. Responses to the 3270 family display devices do not erase what is displayed on the screen and are displayed in the last 40 character positions of the line.

After a message-switching transaction has been started, you can press the CLEAR key if you want to continue from an empty screen. A CLEAR/CANCEL OR CONT xxxxx response is displayed, where xxxxx is either INPUT, ROUTE, or MSG. You can continue the transaction or, if you press the CLEAR key a second consecutive time, you can cancel the transaction. A better method of erasing the screen is to press the ERASE INPUT key, because this does not cause an interaction with the processor.

Because positions on the screen that are not keyed are not received, you can position and display data on the screen in various ways. For example, you can start each option on a new line of the display screen. You must include a comma between consecutive options. Also, if you want to start a new line within the message text you should enter a field mark character to indicate the end of the current line. All field mark characters within a message (including the first character of the message) are interpreted as NL characters (there is no corresponding key on 3270 family display devices) and cause correct positioning of the message on all other terminals.

IBM 3767 Communication Terminal

The keyboard/printer unit of the 3767 is eligible for entering and receiving messages.

Sequential terminal

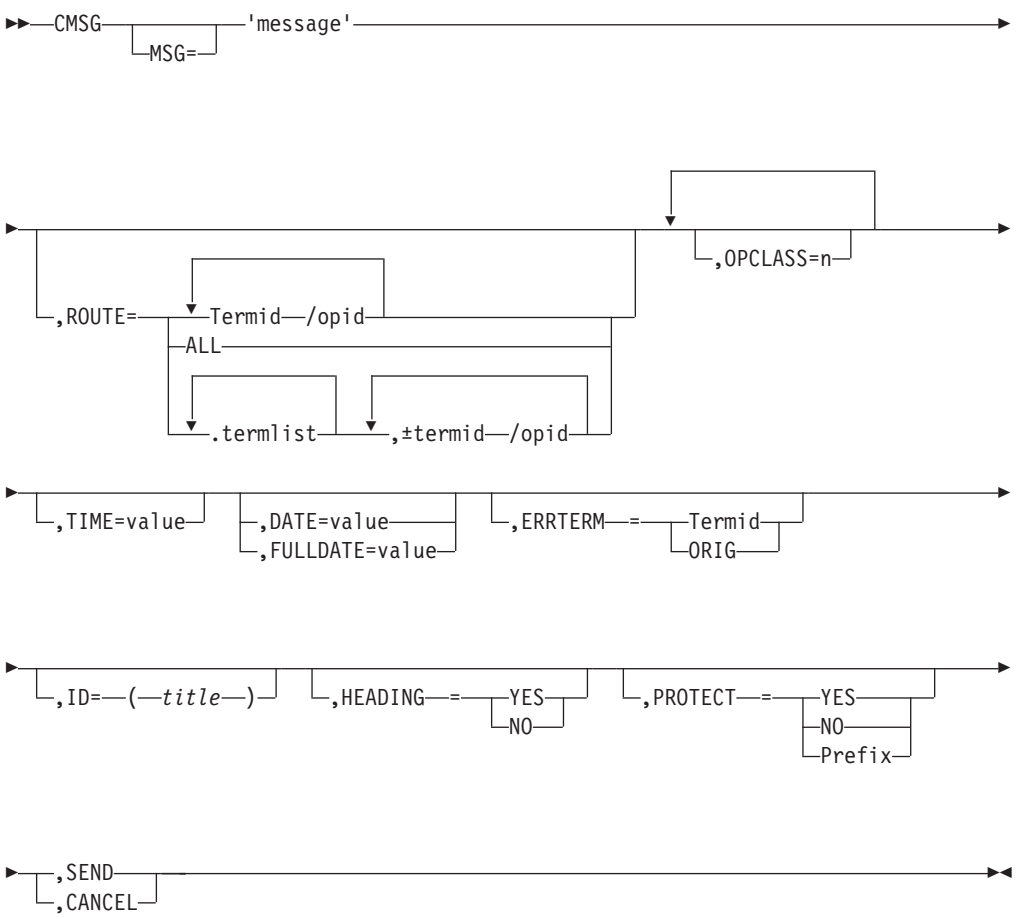
On a sequential (BSAM) terminal, the CMSG message-switching facility remains conversational until a message-switching transaction is completed or canceled.

Variable conditions requiring a response, such as operators not being signed on, can be handled easily from an interactive terminal. These conditions, however, cannot always be anticipated, and the message-switching task continues reading input and issuing error messages until a valid response is received. It is possible that an entire sequential input stream could be consumed in such a process. You must ensure that a CMSG CANCEL record appears in the sequential input following each message-switching transaction that could possibly require an unanticipated reply. This CMSG CANCEL record is advised following the final message-switching transaction.

CMSG command

Describes the syntax and attributes of the CMSG command.

CMSG



CMSG command options

You can use the CMSG command to modify messages sent from your terminal.

The message-switching options are listed below. Except for CANCEL, you can specify the first letter of each option instead of the entire option.

CANCEL
Specifies that the current input is to be ignored and institutes a

non-conversational status between the terminal and the message-switching transaction. CANCEL must be the last 6 characters of the input. CANCEL is also effective within a message.

DATE=value
The date on which you want your message to be delivered. It can be specified in any of the following forms:

yy.ddd
year (00–99) and day (001–366).

mm/dd/yy
month (01–12), day (01–31), and year (00–99).

mm/dd
month (01–12) and day (01–31).

+d number of days (0–4).

The first three of these forms provide ways of specifying absolute dates, with the year (where used) in a 2-digit format. For example, if the current system date is in the year 1997, January 31 1997 could be specified as 97.031, 01/31/97, or 01/31. In this last case, the year of the current system date is assumed to be the year for delivery of the message.

Note: References to "system date", "system time", "system year" and so on, mean the date time or year as would be returned by EXEC CICS ASKTIME.

If DATFORM=DDMMYY was specified in the CICS system initialization parameters, enter the second and third of these as dd/mm/yy or dd/mm.

- If you want to specify an absolute date with the year in a 4-digit format, then use the FULLDATE parameter.

The fourth form allows you to specify a number of days from today. For example, a value of DATE=+3 (or D=+3) means that the message is to be transmitted 3 days from today. The number must be in the range 0–4. DATE=+d entries are not accepted when the system time is 2330 - 0030, (to avoid confusion at or near midnight). If you use this form of the command within 30 minutes of midnight, the following error message is issued:

```
+DATE INVLD FROM 2330 to 0030
```

You can also specify a time for message delivery using the TIME= option, which is described CMSG command options, SEND. The effects of TIME= and DATE= together are as follows:

- If you specify neither a time nor a date, the message is transmitted as soon as the receiving terminal is free.
- If you specify a time but no date, the message is transmitted at the specified time today. For example, if the time now is 09.00 and you specify TIME=0930, or TIME=+30, the message is transmitted at 09.30 today.
- If you specify a date but no time, the message is transmitted at the current system time on the specified date. For example, if the time now is 10.30 and you specify DATE=+2, the message is transmitted at 10.30 in two days' time.
- If you specify both a date and a time, the message is transmitted at the specified time on the specified date. For example, if you specify DATE=07/29/98 and TIME=1130, the message is transmitted at 11.30 on 07/29/98.

Note:

1. In all cases, the delivery time that you request must be less than 100 hours from the beginning of the current day. This means that the delivery time can never be later than 03.59 on the fourth day from the current day.
2. When processing date options entered in the form yy.ddd, mm/dd/yy and dd/mm/yy, CMSG operates a 'sliding 50 year window' to establish whether the year is in this century, the previous century, or the next century. The two-digit year is initially assumed to be in the same century as the current date. If this assumed year is more than 50 years in the past or more than 50 years ahead, it is adjusted accordingly. For example, if today's date is the 31st December 1997, the following DATE options are handled as follows:
   - DATE=99.001 is initially assumed to be the year 1999. Since it is within 50 years of the system year, the year 1999 is determined to be the delivery date for the message.
   - DATE=00.001 is initially assumed to be the year 1900. Since this year is more than 50 years ago, the delivery date is established as the year 2000.

   In both of these examples above, the delivery date is not accepted and the message 'DATE TOO FAR IN FUTURE' is displayed.

   Note that the FULLDATE operand allows a four-digit year to be specified, and removes any possible ambiguity when using the DATE operand.

ERRTERM

"termid" is the identifier of the terminal to which notification is to be sent if the message is purged because it is undeliverable.

ORIG is a way of specifying the identifier of the originating terminal.

Note: A message is considered undeliverable to a destination if it cannot be delivered within a specified interval after the requested delivery time. This interval is specified by the system programmer. If no interval is specified, no action is taken for undelivered messages, and the ERRTERM option has no effect.

If PRGDLAY is specified in the system initialization table (DFHSIT), the transient data destination CSMT is notified of the number of undeliverable messages purged for a terminal. In addition, if ERRTERM is entered, the specified terminal is notified of the message number, title identifier, and destination of the message.

FULLDATE=value

The FULLDATE option is like the DATE option, but it requires a four-digit year to be entered. It specifies the date on which you want your message to be delivered. It can be specified in any of the following forms:

yyyy.ddd
year (0000–9999) and day (001–366).

mm/dd/yyyy
month (01–12), day (01–31), and year (0000–9999).

mm/dd
month (01–12) and day (01–31).

+d number of days (0–4).

The first three of these forms provide ways of specifying absolute dates, with the year (where used) in a 4-digit format. For example, if the current system date is in the year 1997, December 31 1997 could be specified as 1997.365,

12/31/1997 or 12/31. In this last case, the year of the current system date is assumed to be the year for delivery of the message.

(If DATFORM=DDMMYY was specified in the CICS system initialization parameters, enter the second and third of these as dd/mm/yyyy or dd/mm).

The fourth form allows you to specify a number of days from today. For example, a value of FULLDATE=+3 (or F=+3) means that the message is to be transmitted 3 days from today. The number must be in the range 0–4. FULLDATE=+d entries are not accepted when the system time is 2330 - 0030, (to avoid confusion at or near midnight). If you use this form of the command within 30 minutes of midnight, the following error message is issued:

```
+DATE INVLD FROM 2330 to 0030
```

You can also specify a time for message delivery using the TIME= option, which is described in CMSG command options, SEND. The effects of TIME= and FULLDATE= together are as follows:

- If you specify neither a time nor a date, the message is transmitted as soon as the receiving terminal is free.
- If you specify a time but no date, the message is transmitted at the specified time today. For example, if the time now is 09.00 and you specify TIME=0930, or TIME=+30, the message is transmitted at 09.30 today.
- If you specify a date but no time, the message is transmitted at the current system time on the specified date. For example, if the time now is 10.30 and you specify FULLDATE=+2, the message is transmitted at 10.30 in two days' time.
- If you specify both a date and a time, the message is transmitted at the specified time on the specified date. For example, if you specify FULLDATE=07/29/1998 and TIME=1130, the message is transmitted at 11.30 on 07/29/1998.

Note: In all cases, the delivery time that you request must be less than 100 hours from the beginning of the current day. This means that the delivery time can never be later than 03.59 on the fourth day from the current system date.

HEADING

specifies heading information. You can use H or HEADING in place of HEADING=YES.

YES Specifies that the current time, date, and identifier of the originating terminal is to precede the message text.

NO causes a previous heading request to be ignored.

ID=(*title*)

title specifies the title (maximum length 62 characters) to be associated with the message.

See Chapter 32, “CSPG - page retrieval,” on page 619 for commands to request a display of the titles of all messages queued for immediate delivery to that terminal.

MSG=message

“message” is the text of the message to be sent. The keyword MSG and the equal sign are optional. You must enclose the text within single quotation marks. A single quotation mark to be included as part of the message must be represented by a pair of single quotation marks. The message may be continued across multiple consecutive inputs.

If the ending single quotation mark is omitted, the entire input is treated as part of the message and a request to continue the message is sent to the terminal. The entire transaction may be canceled, or alternatively, options previously entered for this transaction may be saved by entering a single quotation mark followed by a comma to terminate the MSG option. The correct message can then be reentered; the previous incorrect message being ignored.

A single quotation mark at the end of data in a MSG option means either the end of the MSG option, or the first of a pair of single quotation marks indicating that a single quotation mark is to be included as part of the message.

In this situation, the response to the terminal is:

```
CONTINUE INPUT OR MSG
```

If the first character of the next input is a single quotation mark, it is treated as the second of a pair of single quotation marks and the message is continued. Any character other than a single quotation mark causes the message to be complete, and that character is treated as the first character of a new option.

New-line (NL) characters within the message are kept. (If the first character is a new-line character, it is deleted.) This allows the operator to enter `M='` and then carriage return (CR) or the equivalent of CR, to begin entering the message text at the left margin. The first CR is deleted. Additional CRs may be entered if blank lines are desired at the top of the transmitted message.

Note: If the HEADING option is specified, these blank lines appear between the heading (time, date, and originator's terminal identifier) and the message.

With NL processing, the delivered message is positioned at the left margin. If an unformatted message, or a line within a formatted message, exceeds the line width defined for the receiving terminal, sentences are split between words for any line exceeded.

OPCLASS

One or more numbers, each of which can be in the range 1–24, that define the operator classes that must be signed on before a message can be delivered. If more than one number is specified, the list must be enclosed within parentheses. For example, OPCLASS=(8,2) causes the message to be sent to all terminals that currently have an operator of class 8 or 2 signed on, and to all terminals that have that operator security value specified in their installed definitions. If OPCLASS=1 is specified, the message is routed to all terminals that are in service, regardless of whether an operator is signed on or not.

If ROUTE is specified as well, the message is routed to all requested destinations, but is not eligible for delivery to a terminal unless the class of the operator signed on matches one of the numbers specified by OPCLASS. However, if a ROUTE destination is qualified by an operator identifier, OPCLASS is ignored for that destination. For more information about how ROUTE= and OPCLASS= are used together, see the description of the ROUTE option.

PROTECT

specifies message recovery for a CICS emergency restart. You can use P or PROTECT in place of PROTECT=YES.

YES Specifies that $$ is to be prefixed to the temporary storage data identifier of the stored message.

NO Specifies that a previous protect request is to be ignored. This is done by using the default prefix of **. The same method is used to omit the option altogether.

prefix
Specifies a 1-or 2-character prefix to be used for the temporary-storage data identifier of the stored message. If a single character only is specified, a $ is provided as the second character. (For example, PROTECT=T causes a prefix of T$.)

If this option is omitted, a default prefix of ** is used. ** is also the default for user application programs issuing BMS message requests where no protection is specified (REQID option omitted).

For each prefix specified in the PROTECT option, you must define the temporary storage model to CICS as recoverable, so that message recovery is effective for that prefix.

ROUTE
Specifies the destinations to receive the message. For routing messages to 3600, 3770 (batch), or 3790 (batch) terminals, see "Examples of 3600 and 3770 batch destinations" on page 588.

Termid
is the identifier or identifiers of the terminals to which the message is to be routed. For example, ROUTE=(LA04,OL,SF2) routes the message to the three terminals with the identifiers LA04, OL, and SF2. If routing is performed to several terminals of the same device and map suffixes, CICS processes the message identically for all of them and the most restricting page size prevailing is used.

The length of the terminal identifier specified in a message-switching transaction must be in the range 1–4 characters, and must not contain any of the following characters:

/	slash
,	comma
)	right parenthesis
(	left parenthesis
+	plus sign
-	minus sign
*	asterisk
	blank.

Note: A single message can be delivered more than once to the same terminal. For example, the instruction ROUTE=(T001,T001) causes two transmissions of a single message to terminal T001. If the destination terminal is in TRANSCEIVE status, the message appears consecutively at the terminal. If the terminal is in TRANSACTION status, the operator must request delivery of the message.

/opid
is a 1-to 3-character operator identifier preceded by a slash. The message is routed to the first terminal at which an operator with that identifier is currently signed on. For example, ROUTE=/PJ routes the message to the first terminal found (and only the first) with the operator identifier PJ currently signed on. If no such terminal is found, the sending operator is notified. The operator identifier that you specify must not contain any of the following characters:

,	comma

) right parenthesis

space.

Termid/opid
Is a terminal identifier qualified by an operator identifier to restrict the message delivery to the specified operator at the terminal location. For example, ROUTE=(LA04,OL/LBS,SF2) routes the message to terminals LA04 and SF2. The message is routed to terminal OL only if the operator whose identifier is LBS is signed on at that terminal.

ROUTE=(T001,T001/OP1,/OP1) causes the same message to be delivered three times to the same destination if the operator OP1 is signed on at T001.

ALL
Causes the message to be broadcast to all terminals.

There is a variable limit on the number of terminals to which a message can be sent. This limit depends on a combination of factors. Significant factors are the types of terminal in use, the number of each type, and the length of message sent. The CMSG transaction is abended with an abend code of ABMC if the limit is exceeded.

Note: If a CMSG ROUTE=ALL is issued to many terminals, a task for each terminal is initiated up to the MAXTASK value. Because the tasks are single threaded, they are suspended and can give rise to an SOS condition. For guidance about avoiding this, see the *CICS Performance Guide*.

.termlist
Is a 1-or 2-character terminal list table (TLT) suffix preceded by a period. For example, .H3 identifies the terminal list table DFHTLTH3. A maximum of 10 terminal lists can be specified, and the terminal lists that you specify are merged together. The entries in the terminal lists contain terminal identifiers, or operator identifiers, or both. Duplicate entries within a single TLT are kept, though entries that are duplicated among the lists are deleted. (Entries are considered duplicate if each has the same terminal identifier and operator identifier.)

Here are two examples that show the effects of merging TLTs that contain duplicate entries. For these examples, assume that terminal list table DFHTLTL1 contains T001 twice, and that DFHTLTL2 contains T001 and T001/OP1.

- If you specify ROUTE=(.L1,.L2), all entries from DFHTLTL1 are included as destinations. Duplicate entries within DFHTLTL1 are kept. All entries from DFHTLTL2 are checked for duplicates against the entries in the previously specified DFHTLTL1 and, if a duplicate is found, it is not repeated.

  The resulting destination list is T001, T001, T001/OP1.
- The order in which you specify the TLTs is significant. If you specify R=(.L2,.L1), the DFHTLTL2 entries T001 and T001/OP1 are included in the destination list. However, the two entries for T001 in DFHTLTL1 are not included because T001 is already in DFHTLT2. In this case, the resulting destination list is T001, T001/OP1.

(±*termid/opid*,...**)**
A +termid/opid adds the specified destination (if not a duplicate) to the destinations contained in the requested TLT. A -termid/opid deletes the specified destination from the requested TLT. A -termid, without an opid,

deletes all destinations of that terminal (with or without operator identifier) resulting from the requested TLT. + or -termid/opid parameters affect only those entries that result from requested TLTs, and have no effect on other + or - termid/opid parameters in the same request. All TLT suffixes must be entered before any + or - parameters.

Here are some examples that show the effects of specifying both TLTs that contain duplicate entries and ± entries. For these examples, assume that terminal list table DFHTLTL1 contains T001 twice, and that DFHTLTL2 contains T001 and T001/OP1.

- ROUTE=(.L1,.L2,+T001) has the same effect as R=(.L1,.L2). The entry +T001 is not added, because it is a duplicate of an entry from DFHTLTL1. The resulting destination is T001, T001, T001/OP1.
- ROUTE=(.L1,.L2,+T001/OP1,-T001) does not add +T001/OP1 because it is a duplicate of an entry in DFHTLTL2. The -T001 causes all entries from TLTs that refer to T001 (regardless of whether they are qualified by an operator identifier) to be deleted. The message ALL ROUTE ENTRIES DELETED is issued.

  If DFHTLTL2 did not contain the entry T001/OP1, the +T001/OP1 instruction would cause that entry to be added to the destination list. The -T001 instruction would not then delete the T001/OP1 entry from the list, because the effects of the + and - instructions are not cumulative: they act in isolation on the original concatenated TLTs.
- ROUTE=(.L1,.L2,-T001,+T001/OP1); the -T001 causes all entries from the TLTs that refer to T001 (including the T001/OP1 entry in DFHTLTL2) to be deleted. The +T001/OP1 entry is then added and becomes the only resulting destination. There is no duplicate because it has just been deleted.

A ROUTE option can be divided across multiple consecutive inputs. However, if it refers to a TLT, it must be completed in the same input in which it was started. An individual ROUTE parameter (termid/opid) cannot be split across two inputs.

When both ROUTE and OPCLASS are specified together, OPCLASS further restricts the message transmission. For example, ROUTE=(LA04/PJL,/MGK,OL), OPCLASS=4 routes the message to terminal LA04 if the operator whose identifier is PJL is signed on. The message is also sent to the first terminal with the operator whose identifier is MGK signed on. An operator whose class is 4 must be signed on to OL before the message can be routed there. Note that the OPCLASS value is acted on only when no operator identifier is specified.

SEND
Specifies that all of the options have been entered and that the message is to be routed. SEND is the final option and must be followed by a space or an end-of-data.

TIME=value
"value" is the time at which you want the message to be delivered. You can specify the time in one of the following four ways:

hhmm
Where "hhmm" is an absolute time in the range 0001–2400. For example, TIME=1145 causes the message to be transmitted at 11.45 am. The minutes value must be less than 60.

+hhmm
Where “hhmm” is the number of hours and minutes from the current time. The minutes value must be less than 60. For example, TIME=+0720 means that the message is to be transmitted in 7 hours and 20 minutes from now. A value of TIME=+2400 means the same as DATE=+1.

+mm Where “mm” is the number of minutes from the current time. This value must be in the range 0–99. So, for example, a value of TIME=+75 causes the message to be transmitted 1 hour and 15 minutes from now. The values TIME=+90 and TIME=+0130 both cause the message to be transmitted in 90 minutes time.

+m Where “m” is the number of minutes from the current time. This value must be in the range 0–9. So, for example, a value of TIME=+5 causes the message to be transmitted 5 minutes from now.

If you specify a delivery time on the current day that falls within the past hour, it is interpreted as a request for immediate delivery. An earlier time than that is considered already passed and is treated as an error. The following message is issued:

```
TIME ALREADY PASSED
```

Note that, if the current time is 00.15, T=2345 is interpreted as 23.45 today because there has been a change of date. The message is not therefore transmitted immediately.

Logical unit destinations

You can use the CMSG transaction to send messages to logical units. Each logical unit in a CICS-SNA network is identified by a single terminal identifier and, if the logical unit is capable of receiving message text, messages can be routed to it in the same way as they are routed to non-SNA terminals.

You can also route messages by operator identifier if the logical unit supports operator signon. You can specify SNA and non-SNA destinations in the same ROUTE option.

The destination for a message sent to a logical unit can be a display or printer device, a data set, or an application program in a subsystem controller. To the message sender, the destination behaves like a “terminal”, and any necessary formatting is performed by the CMSG transaction or within the subsystem controller itself.

Logical device codes

Some types of logical unit (LU) can be used to get access to more than one resource within a subsystem.

For example, data sent to a 3601 LU might be intended for an IBM 3604 Keyboard Display, an IBM 3618 Administrative Line Printer, or some other element of the IBM 3600 Finance Communication System.

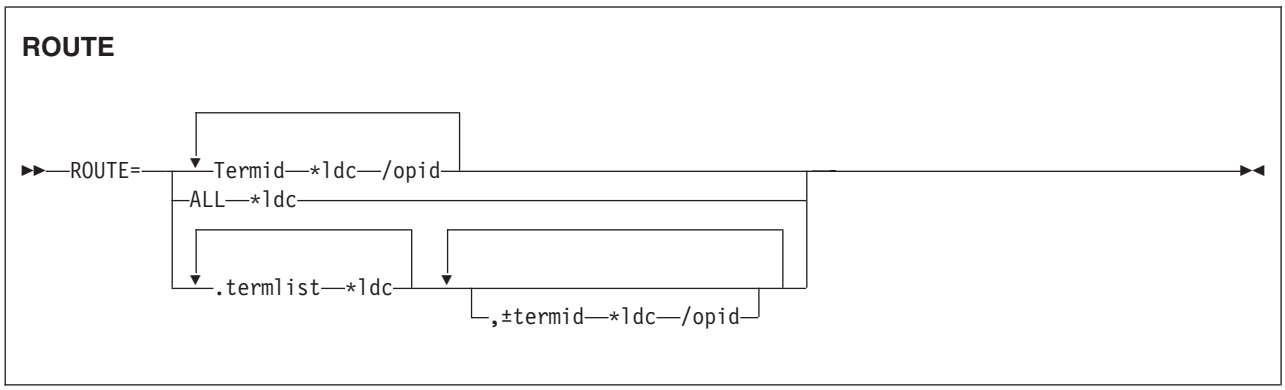
The facility provided by CICS to permit destination selection within LUs of this type is the logical device code (LDC).

The LUs that support destination selection by LDC are:

- 3601 LU
- 3770 batch LU

- 3770 batch data interchange LU
- 3790 batch data interchange LU.

For the user of the message-switching transaction, the LDC is a 2-character mnemonic code whose meaning is defined by the CICS installation. It can be used to qualify an LU destination by including it in the ROUTE option in the syntax diagram that follows:
where:



***ldc**
is a 2-character LDC mnemonic preceded by an asterisk (*) that qualifies the destinations. The *ldc parameter can qualify an LU identifier (termid), a general broadcast (ALL), or a terminal list table specification (.termlist). The *ldc parameter applies only to LUs, not to any start-stop or BSC terminals.

Different LDC mnemonics can be included in one ROUTE option specification; however, all destinations for one message must indicate the same device type.

Termid*ldc
associates an LDC mnemonic with an LU identification.

ALL*ldc
is a general broadcast to all terminals with the same LDC mnemonic qualifying all LUs.

Termlist*ldc
.termlist*ldc
qualifies all entries in this terminal list table with the specified LDC mnemonic. This overrides any LDC specified within the terminal list table. This LDC specification does not apply to start-stop or BSC terminals.

Note: This qualification of a TLT occurs before any succeeding TLTs or +/- entries are processed, see example 9 in "Examples of 3600 and 3770 batch destinations" on page 588.

+termid*ldc/opid
adds a destination, if not a complete duplicate to any contained in the requested TLTs.

-termid*ldc/opid
deletes duplicate destinations resulting from the requested TLTs. A -termid*ldc, without an opid, deletes all destinations of that termid*ldc (with or without operator identifiers) resulting from the requested TLTs. A -termid/opid, without an LDC mnemonic, deletes all destinations of that termid/opid (with or without LDC mnemonics) resulting from the requested TLTs.

If a destination is specified by /opid without termid, it becomes termid/opid, in which termid is the identifier of the first terminal, or LU to which an operator with that identifier is currently signed on. If no such terminal or LU is found, the destination is not valid and the operator is notified.

Examples of 3600 and 3770 batch destinations

These examples assume the following:

- Terminal identifiers T36n and T37n are 3600 logical units and 3770 batch logical units respectively; all others are start-stop or BSC terminals.
- LDC mnemonics DS and LP have been defined for the system and T361, T362, and T363 during resource definition.
- LDC mnemonic P1 has been defined for the system and T371, T372, and T373 during resource definition.

  You can also use the TYPETERM LDCLIST attribute of RDO to define these mnemonics instead of the DFHTCT TYPE=TERMINAL macro shown.
- Terminal list table DFHTLTL3 contains entries for:

  T361*DS
  T362*DS/OP1
  T362*DS/OP2
  T363
  T371*P1
  T372*P1/OP1
  T372*P1/OP2
  T373
  T40.
- Terminal list table DFHTLTL4 contains entries for:

  T361*DS
  T362*DS/OP1
  T362*DS/OP2
  T363
  T40.

1. R=T361*DS
Route message to terminal identifier T361 qualified by LDC mnemonic DS.*2. R=(T361*DS,T362*DS/OP1,T363,T371*P1, T372*P1/OP1,T373,T40)* Route message to:

- T361 and T362 qualified by LDC mnemonic DS
- T363, using the default LDC
- T371 and T372 qualified by LDC mnemonic P1
- T373 console keyboard/printer
- T40, which is a start-stop or BSC terminal.

T362 and T372 require that an operator with identification OP1 be signed on before the message can be sent.

Note: The default LDC mnemonic for T363 must indicate the same device type as LDC mnemonic DS.

3. R=ALL*LP
Route message to all terminals (3600, 3770 batch and interactive logical units, start-stop, and BSC) with all 3600 destinations qualified by mnemonic LP. LP is ignored for start-stop and BSC destinations.

4. **R=.L3**
   Use the terminal list table DFHTLTL3 for message destinations. (This is the same as example 2, plus T362*DS/OP2 and T372*P1/OP2.)

5. **R=(.L3,-T362*DS/OP2,-T372*P1/OP2)**
   This is the same as example 4, but deletes T362*DS/OP2 and T372*P1/OP2, so is the same result as example 2.

6. **R=(.L3,-T362*DS)**
   This is the same as example 4, but deletes all entries for T362*DS (with or without opids). The result is T361*DS, T363, all T37n terminals, and T40.

7. **R=(.L3,-T362)**
   This is the same as example 6. –T362 deletes all entries for T362.

8. **R=.L4*LP**
   LDC mnemonic LP qualifies (overrides) all entries in DFHTLTL4. Resulting destinations are: T361*LP T362*LP/OP1 T362*LP/OP2 T363*LP T40*LP

   Note: The LP mnemonic has no effect on the start-stop or BSC terminal T40.

9. **R=(.L4*LP,-T362*DS)**
   The –T362*DS causes no action, because the TLT destinations are qualified by LDC mnemonic LP before the additions or deletes are processed, thus causing no matching entry to delete.

10. **R=(.L4,+T363*LP)**
    Cause error message 'INVALID LDC AT T363*LP' to be generated. LDC mnemonic LP has a different device type from LDC mnemonic DS (first 3600 destination encountered is T361*DS). All 3600 destinations for one message must indicate the same device type. All 3770 batch logical unit destinations for one message must indicate the same device type.

11. **R=/OP2**
    Route message to the first terminal or logical unit found in the CICS terminal definition with operator identifier OP2 currently signed on. If OP2 is signed on to T362, the resulting destination is T362/OP2 with the default LDC mnemonic for logical unit T362. The default is DS because it is the first LDC mnemonic defined for T362. The resulting destination is T362*DS/OP2.

Related concepts

Chapter 27, “CMSG - message switching,” on page 573
Use CMSG to send messages from your terminal to one or more destinations.

“Subsystems and terminal systems” on page 577
You can use the CMSG message-switching facility ROUTE option to manage responses from display devices, communication terminals, and sequential terminals. Variable conditions requiring a response cannot always be anticipated, and the message-switching task performs this function by reading input and issuing error messages until a valid response is received.

Related tasks

“Sending a message” on page 573
To send a message to another terminal or terminals, enter the transaction identifier, CMSG, followed by the applicable options.

Related reference

“CMSG command” on page 578
Describes the syntax and attributes of the CMSG command.

“CMSG examples” on page 597
These examples show you how to modify the behavior of CMSG by changing the parameters.

Message-switching responses

Every input from the terminal receives a response. Most responses are preceded by indicators where the letters before OK represent the first character of each of the CMSG options (except CANCEL). These indicators identify the options that have been processed and that are currently in effect.

The first character of each CMSG option is as follows:

```
D    DATE
E    ERRTERM
H    HEADING
I    ID
M    MSG
O    OPCLASS
P    PROTECT
R    ROUTE
S    SEND
T    TIME
```

Errors may occur because of:

- Syntax (for example, misspelled option, unbalanced parentheses, terminal identifier more than 4 characters, invalid option separator, and message and destination not provided).
- Specification (for example, the specified terminal has not been defined to CICS).
- Operation (for example, operator not currently signed on to the system).

Syntax errors within an option cause it to be rejected by the message-switching routine. To correct a known error, reenter the option before typing the SEND keyword.

Syntax errors cause no data to be processed in that input beyond the detected error. All following data must be reentered to be included.

A specification or operation error message may be written to the terminal with an indicator that shows that the option in question has been processed and is still in effect. Two possible error messages are:

1. M OK TERMID GT 4 CHAR AT T004A

   This message has been issued because the terminal identifier is greater than 4 characters. The absence of the R indicator preceding the message shows that ROUTE is not currently in effect and that the message must be reentered.

2. MRHS OK INVLD TERMINL ID T004

   This message has been issued because the specified terminal is not known to CICS. The R indicator shows that ROUTE is still in effect. Operator action as a result of this type of message is either:

   - A space (entry deleted)
   - Termid/opid specifying a termid, or operator identifier, or both (entry changed)
   - Any option (the route option or another option may be entered).

   Note: If an option is entered, the error message is repeated unless a new ROUTE option is entered before SEND. If the incorrect entry is changed or deleted by responding with a space or termid/opid, the message is routed unless other errors are still present. These errors are indicated by another message. If an option is entered, it signifies normal entry mode in which multiple options can be entered requiring another SEND option to end the input.

The following is a list of all possible responses from the message-switching program (DFHMSP). Because of device-dependent considerations, all responses are limited to 25 characters plus the indicators.

ALL ROUTE ENTRIES DELETED
A space entered in reply to a incorrect destination message caused the remaining destinations to be deleted, leaving no destinations to which to route the message. This condition forces a return to normal option mode, and a ROUTE option must be reentered to provide the message destinations.

CANCELED - TEMP STRG ERR N
N has a value in the range 1–4.

- N=1. An error occurred on a temporary storage (TS) request.
- N=2. An error occurred on a temporary storage (TS) request. A possible cause of this response, not related to a TS problem, is a request for a TS record that does not exist. This can happen if:
  - The transaction identifier is entered followed by a C. This extended transaction identifier (CMSGC) normally requests a continuation of a message-switching transaction, but no previous status information exists for this terminal. See “Message-switching continuation” on page 575.

    Note: A CICS cold start of temporary storage causes the loss of any status saved from a previous CICS process.
  - The transaction identifier is entered in lowercase on a 3270 family display device that is defined as a dual case keyboard, and for which no previous message-switching transaction status exists.

    Note: The transaction identifier in the terminal input/output area (TIOA) is not translated. To correct this problem, enter the transaction identifier in uppercase.

N=2 also occurs when transaction routing is being used and indicates that the transaction identifier used to initiate the task is different to that specified in the CICS system definition.

- N=3. An error (most likely caused by TS) occurred on a DFHBMS TYPE=TEXTBLD request.
- N=4. An error (most likely caused by TS) occurred on a DFHBMS TYPE=PAGEOUT request.

CLEAR/CANCL or cont input|route|msg
An operator on a 3270 family display device pressed the CLEAR key and the message states continue (CONT):

INPUT
The last option in the previous input is complete. Enter an option to continue.

ROUTE
The last option in the previous input is ROUTE and a closing parenthesis cannot be found. Continue entering ROUTE destinations.

MSG
The last option in the previous input is MSG, and a single quotation mark to end the message cannot be found, or a single quotation mark is found as the last character of the input, and might be the first of a pair of single quotation marks. If the previous input does not end with a single quotation mark, continue entering the message. If the previous input ends with a single quotation mark, enter another single quotation mark to continue the message (which causes a single quotation mark to be included), or enter any option that causes the message to be considered complete.

Note: If the operator replies by again pressing the CLEAR key (two consecutive times) it has the same effect as entering CANCEL for the last 6 characters of the input.

CONTINUE INPUT
A SEND option has not been processed, and a ROUTE or MSG option was not being continued. Continue by entering any option.

CONTINUE INPUT OR MSG
The previous input ended with a single quotation mark with the MSG option being processed. This single quotation mark could specify the end of the MSG option or could be the first character of a pair of single quotation marks (signifying that a single quotation mark was to be included within the message).

If the next input starts with a single quotation mark, the message is continued with a single quotation mark included in the message. A nonpaired single quotation mark is required to end the MSG option.

If the next input starts with any character other than a single quotation mark, it is treated as the beginning of a new option and the MSG option is considered complete.

CONTINUE MSG
The previous input ended with the MSG option being processed, but no single quotation mark ended the entered text. Continue the message.

CONTINUE ROUTE OPERAND
The ROUTE option was being processed in the previous input and a space or an EOD was found. Continue the ROUTE option.

DATE ALREADY PASSED
A delivery date before the current day is invalid.

DATE INVALID AT xxxxxxxx
DATE must be in one of the following forms:

```
yy.ddd      year (00–99) and day (001–366)
mm/dd       month (01–12) and day (01–31)
mm/dd/yy    month (01–12), day (01–31),
                           and year (00–99)
D=+d        day (0–4)
```

DATE NOT xx/xx at xxxxx
DATE must start with 2 digits followed by a slash (for month), or a period (for year). This message is issued if the third character of DATE is not a slash or a period.

Note:

1. September 5, for example, must be specified as 09/05 and not as 9/5.
2. The date must be valid, not 06/31 for example.

DATE TOO FAR IN FUTURE
Future delivery may not be specified after 0359 hours on the fourth day from the current day. The error is either:

1. D=+n, where n is greater than 4, or
2. DATE specified explicitly as being after the fourth day from the current day, or
3. A condition of DATE and TIME that exceeds the stated limit.

If conditions 1 or 2 occur, the DATE option is not in effect (the D status indicator is not present). If condition 3 occurs, both the DATE and TIME options are currently in effect (the D and T status indicators precede the message), and one or both of them must be changed to a valid delivery time or the error occurs again.

ERRTERM INVALID AT xxxxx
An ERRTERM must be 1–4 characters.

ERRTERM INVLD/UNSUPP xxxx
A terminal identifier specified for the ERRTERM option does not have an entry specified in the CICS terminal definition, or it is for a terminal type not supported by BMS, and is therefore invalid. Valid replies by the operator to this are:

1. A terminal identifier or ORIG in the same format as a normal ERRTERM destination (without entering the option ERRTERM or E). The message is routed following this input unless the new ERRTERM specified is invalid. In this case, the message is repeated.
2. A space as the first character (followed by an EOB or comments). This causes the entire ERRTERM specification to be eliminated. The message is routed following this input.
3. Any option that causes a return to normal option processing mode. If another ERRTERM option is not entered to correct this condition, the same error message is repeated after SEND has been reentered.

HEADNG NOT YES/NO AT xxxx
A HEADING option may only be specified as follows:

```
HEADING
HEADING=YES
HEADING=NO
H
H=YES
H=NO
```

ID OPERAND GT 62 CHARS
An ID option must be from 1–62 characters in length, and enclosed within parentheses.

ID OPERAND MISSING
The ID option requires the identifier to be enclosed within parentheses.

ID PARAM MISSING OR NO)
The ID option must be from 1–62 characters in length and enclosed within parentheses.

INVALID LDC AT xxxxxxxx
The LDC mnemonic is not valid for the logical unit, or a previous entry contained an LDC mnemonic that indicated a different device type.

Note: A missing LDC mnemonic can cause this message if the default mnemonic for that station indicates a different device type than a previously indicated mnemonic (specified or defaulted).

INVALID OPND AT xxxxx
An option xxxxx is invalid (possibly caused by a keying error). No data beyond this point was processed. The status indicators represent the options that are in effect.

INVALID TBL SFX AT xxxxx
A terminal list table suffix must be 1 or 2 characters preceded by a period.

INVLD OPERATR ID AT xxxxx
An operator identifier must be 1, 2, or 3 characters preceded by a slash.

INVLD SEPARATOR AT xxxxx
The end of an option must be followed by a space, a comma (except for SEND), a carriage return (new-line character), or end of data (EOD).

INVLD TERMINL ID xxxx
There is no entry in the CICS terminal definition for the terminal identifier specified for a ROUTE destination. The valid operator reply is one of the following:

1. A terminal identifier, or operator identifier, or both, in the same format as a normal termid/opid ROUTE destination (without entering the option ROUTE= or R=) that replaces the invalid destination.
2. A space as the first character, which causes the invalid destination to be deleted.
3. Any option that causes a return to normal option processing mode. If another ROUTE option is not entered, the same condition exists after the SEND option is reentered.

Note: 1 and 2 cause the message to be routed if the resulting destinations are valid; otherwise, another error message is issued.

LDC NOT 2 CHARS AT xxxxx
The LDC mnemonic that follows the * in the ROUTE destination must be 2 characters.

MESSAGE HAS BEEN ROUTED
A SEND option has been processed and the message has been routed to its destinations. This is the normal completion of a message-switching transaction.

MSG NOT SENT - NO RECEIVER
This condition is caused by an OPCLASS option specified without a ROUTE option and no operators, with the specified operator class, or classes, currently signed-on to terminal types supported by BMS. To correct this situation, a new destination must be entered by the ROUTE option, or OPCLASS option, or both.

NEED OPCLASS OR ROUTE
The SEND option was entered without specifying any destinations for the message. To correct this situation, a destination must be entered by the ROUTE option, or OPCLASS option, or both.

NO INPUT - REENTER
A transaction identifier was entered followed by end-of-data (EOD), or the number of backspace characters entered caused the deletion of all the data (including the position following the transaction identification, if present). If the input does not start with the transaction identifier, the terminal remains in conversation. Otherwise, it is not in conversation.

NO MESSAGE PROVIDED
The SEND option was entered without specifying the message. To correct the situation, provide the message.

NO MSG STRT QUOT AT xxxxx
A MSG parameter must be enclosed within single quotation marks. A single quotation mark must follow an M or MESSAGE.

NO TRMID FOR LDC AT xxxxx
The destination started with an asterisk *. A logical unit identification must precede the LDC mnemonic.

NO TERM LIST TBL-xx
An entry for a terminal list table with suffix xx (DFHTLTxx) was not found in the CICS system definition.

OPCLASS NOT 1-24 AT xxxxx
An OPCLASS parameter must be in the range 1–24.

OPERATORS NOT SIGNDON nnn
The ROUTE option specified operator identifications without terminal identifiers (that is, ROUTE=/opid). nnn operator identifications are not currently signed on.

The valid reply to this is to enter one of the following:

1. A space as the first character, which deletes the nnn destinations.
2. The character D (or any other character) which causes each of the nnn destinations to be individually displayed through the message OPRTR ID NOT SIGNDON.
3. Any option that causes a return to normal option processing mode. If no further ROUTE options are entered, this same condition exists if an operator does not sign on before the SEND option is reentered.

Note: This message is only issued if nnn is greater than 5; otherwise, each destination is individually displayed. Reply 1 causes the message to be routed if the resulting destinations are valid; otherwise, another error message is issued.

OPRTR ID NOT SIGNDON /xxx
An operator with identifier xxx, specified as a destination (entered either by the terminal operator or included as an entry in a TLT whose suffix was specified), is not currently signed on to any terminal controlled by CICS.

The valid reply to this is to enter one of the following:

- A terminal identifier, or operator identifier, or both in the same format as a normal termid/opid ROUTE destination (without entering the option ROUTE or R), which replaces this /xxx destination.
- A space as the first character, which causes the /xxx destination to be deleted.
- Any option that causes a return to normal option processing mode. If no further ROUTE options are entered, this same condition exists if an operator with identifier xxx does not sign on by the time the SEND option is reentered.

Note: The first two replies cause the message to be routed if the resulting destinations are valid, otherwise, another error message is issued.

PROTECT OPND NOT 1-2 CHAR
The PREFIX option of the PROTECT option must not specify more than 2 characters.

ROUTE INVALID AT xxxxx
A field separator entered before the first ROUTE parameter, or 2 consecutive field separators were found. Reenter a valid ROUTE option.

SPACE MUST FOLLOW TRANSID
One or more spaces must follow the transaction identifier before the first option can start. (The character C causes a continuation from the last input preceding a normal completion or a cancelation.)

TERMID GT 4 CHAR AT xxxxx
A terminal identifier must be 1–4 characters.

TERMID NOT +/- at xxxxx
All TLT suffixes must be specified before any individual destination (which must be preceded by a + or -) may be entered. A ROUTE option that specifies a TLT suffix may not be continued on a succeeding input. If a ROUTE option includes more than one parameter, it must be enclosed in parentheses.

TERMINATED BY OPERATOR
The operator entered CANCEL as the last 6 characters of the previous input, or pressed the CLEAR key twice in succession on an IBM 3275 (or 3277) display station. The previous input is canceled and the terminal is **not** in conversational mode with CICS message switching.

TIME ALREADY PASSED
A delivery time for the current day, more than one hour before the current time is considered already passed and is invalid.

If the message is for immediate delivery, enter T=+0, to replace the TIME option that is still in effect.

If the message is for delivery at the specified time on a future day, enter the appropriate DATE option. The specified time is still in effect.

If the message is for delivery at a different time, reenter the TIME option.

TIME INVALID AT xxxxx
TIME must be one of the following:

hhmm
0001–2400 (mm<60)

+hhmm
0000–2400 (mm<60)

+mm 00–99

+m 0–9

where hh=hours and mm=minutes.

TIME NOT 4 CHARS AT xxxxx
A TIME parameter must be a 4-digit number if it is not preceded by +.

TOO MANY TBL SFX AT xxxxx
A maximum of 10 TLT suffixes may be specified.

UNBALANCD PARENS AT xxxxx
A ROUTE or OPCLASS option started with a parenthesis but did not have one at the end, or ended with a parenthesis but did not have one at the beginning. xxxxx represents the characters near the relevant end of the option.

UNSUP OPRT TRMNL xxxxxxxx
An operator identifier without a terminal identifier was specified as a ROUTE destination; however that operator is signed on to a terminal that is a type not supported by BMS and is, therefore, an invalid destination.

The valid reply to this is the same as for the message OPRTR ID NOT SIGNDON.

UNSUP TERMINAL xxxxxxxx
A terminal identifier, specified as a ROUTE destination, is for a terminal type not supported by BMS and is, therefore, an invalid destination.

The valid reply to this is the same as for the message INVLD TERMINL ID.

ZERO LENGTH MSG AT ''
A message must have at least one character enclosed in single quotation marks. A single quotation mark to be included as part of a message must be entered as two separate single quotation mark characters. A message must start with a single quotation mark and may be followed immediately by pairs of single quotation marks.

+DATE INVLD 2330 TO 0030
A DATE=+ option is invalid from 30 minutes before to 30 minutes after midnight.

CMSG examples

These examples show you how to modify the behavior of CMSG by changing the parameters.

Here are some typical examples of the CMSG transaction. Examples 1, 2, 3, and 9 are possible candidates for inclusion in a sequential (BSAM) terminal input stream during startup.

In these examples, # represents a new-line character.

1. CMSG R=ALL,'GOOD MORNING#TODAY IS FRIDAY FEB 1',S

This sends a good morning message to all terminals.

2. CMSG 'GOOD MORNING',R=.G1,T=0900,S

This routes a good morning message to be delivered at 0900 hours to all destinations contained in DFHTLTG1. (Locations in Eastern time zone.)

3. CMSG 'GOOD MORNING',R=(.G2,+CHI4),T=1000,S

This message is for delivery at 10.00 hours to terminals in Central time zone (DFHTLTG2). Also included as a destination is termid CHI4, which was recently added but not yet included in the TLT.

4. CMSG 'SUPERVISORS' 'MEETING IN 5 MINUTES',O=(4,7),S

This routes the message to all supervisors who are assigned operator class 4 or 7 and are currently signed on to CICS.

5. CMSG R=ALL,'NEW PRICE MODEL 402 - $28.70',S

This sends the price of the new model to all terminals.

6. CMSG 'FRED - WHAT IS CURRENT STOCK ON P/N 4837-LP4# BPK',R=BLD3,H,S

This routes a message to BLD3 asking Fred a question and including the initials of the requestor (operator identifier). A heading is included so that Fred also knows the originating termid.

7. CMSG 'PNO 4837-LP4, QTY 26?',R=/BPK,H,S

This shows Fred's answer routed to the terminal at which operator identifier BPK is currently signed on.

8. CMSG 'BRID KADDER ARRIVING 3:15 PST. PLEASE MAKE RESERVATION# M. KRN',R=PADC,H,S

This makes a request for a reservation for a Mr Kadder.

9. CMSG 'SHUTDOWN IN 5 MINUTES',R=ALL,T=1655,S

This sends the shutdown message to all terminals.

Chapter 28. CPIA - unit of work resynchronization transaction

To avoid a warm restart, use the CPIA transaction to resynchronize Web Services Atomic Transaction (WS-AT) units of work and to remove shunted units of work.

To run the CPIA transaction, enter the **CPIA** command in the terminal.

© Copyright IBM Corp. 1982, 2011

Chapter 29. CREA - create REQUESTMODELs for enterprise beans

Use CREA to generate REQUESTMODEL definitions. The transaction can create definitions dynamically, or write them to the CSD, or both. By default, users who are authorized to use CEDA can use CREA.

CREA is not a general tool for CICS resource definition, nor is it a CSD management tool, except in the restricted sense of the particular objects on which it is designed to operate. These are identified in the description of CREA function that follows.

This section contains the following topics:

- "Function:"
- "Using CREA" on page 602
- "CREC" on page 611

Function:

CREA offers you the following functions in turn:

1. displays a list of beans and bean methods within the JAR file related to an installed DJAR,
2. enables the user to associate transaction IDs with those beans and methods,
3. generates list of the REQUESTMODELs, which best match the tranIDs with the beans and methods in the JAR file,
   - Not every association leads to a REQUESTMODEL definition being generated,
   - CREA produces an optimal list of required REQUESTMODEL definitions.
   - Some of the associations lead to the generation of REQUESTMODEL definitions with names that include wildcard characters.
4. presents each REQUESTMODEL in that list to you, with the option to:
   - install the REQUESTMODEL in CICS,
   - define the REQUESTMODEL to the CSD,
   - both install and define, or
   - ignore, (or skip) the REQUESTMODEL.
5. Ends by presenting to you a list of the REQUESTMODEL definitions that you have installed and those that you have defined.

   CREA includes in this list any REQUESTMODELs installed in CICS that are not used by the DJAR (but refer to beans within the DJAR), and also reports on any tranIDs which are used but not currently installed.

Note:

1. CREA only works on installed definitions,
   - You can choose which DJAR definitions are examined by CREA by choosing which definitions are installed at the time that you invoke CREA. In this way you can avoid having CREA present large amounts of information relating to DJARs that do not concern your current activity.
2. Any REQUESTMODELs already installed into CICS are analyzed and tranIDs defined on those REQUESTMODELs are matched to any appropriate beans or methods,

© Copyright IBM Corp. 1982, 2011

Using CREA

About this task

CREA is defined in the DFHADST group and is automatically installed. Your CICS system must be set up to run Language Environment conforming C applications and Java.

CREA presents you with a series of screens as it progresses through its functions.

Note: Choose a suitable code page to connect to your CICS system. If you don't, you'll see the square bracket characters **[** and **]** displaying strangely. One suitable code page is 1047 (United States English).

Starting CREA

Procedure

1. Start the transaction by entering `CREA` (optionally followed by the name of a DJAR definition), on the command line of the screen.
2. Press ENTER.

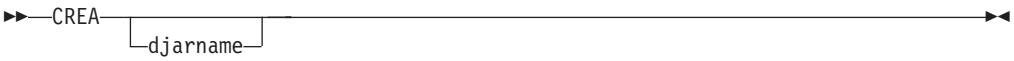
Results

If the DJAR name supplied is valid (that is: if it exists in CICS and is in a resolved state), you are shown the TranID Specification screen (Figure 143 on page 604), which presents the functions available within CREA.

If you do not give the name of a DJAR definition, or if the DJAR definition you name is invalid, you are shown the initial screen (Figure 142 on page 603) and a message indicating the state of the DJAR, or indicating that the DJAR was not found, (as is appropriate).

Command syntax

CREA



Command options

djarname
specifies the identifier of a DJAR definition that you want to work with. The names of the beans and methods in the JAR file associated with this DJAR definition are displayed alongside the tranIDs associated with each.

Response to CREA command

```
   CREA   REQUESTMODEL Definition Transaction           APPLID

   Please enter the name of a DJAR you want to work with.

   DJAR name: ________


   If you want to restrict the display of beans and methods within the
   DJAR to those associated with a particular transaction then you can
   use the tranID filter.  The * character can be used as a wildcard.


   TranID filter: ____



<Message Line >
 PF1=Help    3=Quit
```

Figure 142. CREA transaction: initial screen

If you enter CREA with a valid DJAR name as a parameter, the initial screen (Figure 142) is not presented to you. Instead you are taken directly to the TranID Specification screen (Figure 143 on page 604).

If you enter CREA with an invalid DJAR name as a parameter, the name you entered appears in the DJAR name field of the initial screen Figure 142.

If the status of the named DJAR definition is **resolving**, you have the opportunity to continue. To do so, press enter again and, if the DJAR is now resolved, you are taken to the TranID Specification screen (Figure 143 on page 604).

This initial screen also offers you the opportunity to enter a tranID with which to filter the beans and methods displayed. The filter can be blank to display all beans and methods, or contain a string with optional wildcard characters ('*'). If the filter field is not blank, only those beans and methods that match the filter are displayed.

TranID Specification

When a valid DJAR name has been entered, the TranID Specification screen is displayed. It shows tranIDs alongside the bean and method names that are defined in the deployment descriptor of the JAR file associated with the chosen DJAR definition.

```
CREA  REQUESTMODEL Definition Transaction           APPLID

Associate a transaction ID with a bean or bean method.

DJAR name: SampleEJB                          TranID filter: ____

TranID  Bean/Method/Parameters


CIRP     CICSSample
CIRP >      getCustomerInfo
               int
CIRP >      getEJBHome (inherited from EJBObject)
CIRP >      getHandle (inherited from EJBObject)
CIRP >      getPrimaryKey (inherited from EJBObject)
CIRP >      isIdentical (inherited from EJBObject)
               javax.ejb.EJBObject
CIRP >      remove (inherited from EJBObject)



<Message Line>
 PF1=Help  3=Quit    5=Create ReqMods                      9=Hide Methods
```

Figure 143. CREA transaction: TranID Specification screen

The example given in Figure 143 shows the methods that were found in the Deployment Descriptor of the JAR file associated with DJAR definition *SampleEJB* .

The tranID that is to be used to execute requests (to be identified at run time by an installed REQUESTMODEL resource definition) precedes each of the beans and methods. If there are any currently installed REQUESTMODELs that relate tranIDs to any of the beans or methods in the JAR file, the REQUESTMODEL tranID is shown next to the relevant bean or method. In this example, it is the default value CIRP in every case.

The way that the tranID is determined is this:

- If the *method* already has a tranID relationship, either supplied by you during a previous use of CREA, or defined by an installed REQUESTMODEL, it is displayed, otherwise,
- If a *bean tranID* is known, or if you supply one, any methods on that bean default to the bean tranID.
- For all remaining beans and methods, the Tranid of the default REQUSTMODEL is shown.
- If no default is defined, a default tranID of CIRP is shown.

The rules that influence the use, or not, of default values are :

- After you have entered a tranID, or it has been defined by an installed REQUESTMODEL, it is not replaced by a default value.
- However, if you change a *bean* tranID, all *method* tranIDs for that bean, that have taken a default value, change to match the bean tranID.
- If you change a tranID to spaces, the tranID takes the default value of the bean tranID, and subsequently reacts to any relevant changes to the default, as appropriate.
- If you change the tranID of a method to match that of a bean, then the method tranID ceases to be an absolute value, but becomes one which takes the default of the bean tranID. It subsequently reacts to any changes to that default.

You can see which tranIDs take default values if the bean tranID changes, because:

- The tranIDs that are explicitly defined for beans or methods (that is, they relate to beans or methods because of an installed REQUESTMODEL, or they were entered by you) are referred to as defined tranIDs. They are displayed in white.
- TranIDs that take default values (either CIRP, the overriding default REQUESTMODEL tranID, or the bean tranID defaulting for each of the bean's methods) are displayed in green and are further highlighted with a > character two characters to the right of the tranID entry field.

When entering tranID values:
- an ' sign means - use the value from the tranID field for the bean or method on the line above, and if that is also an ' sign, use the value above that, and so on.

  An ' sign is not valid in the first tranID field displayed. If one is used there, and in any consecutive fields, then those tranIDs are not changed from their current setting.
- a \ sign in the tranID field of any line, causes the display to scroll, making this line the top line on the screen. Only the first \ from the top of the present display is honoured.The tranID field into which the \ is entered is unchanged and is refreshed with its original contents.

At anytime during the process of assigning tranIDs, you can change the tranID filter. The filter causes the display to list only those beans and methods that match the filter. The filter can be blank to show all methods, or can contain a string with optional wildcard characters ('*'). Changes to the filter do not cause associations between tranIDs and beans and methods to be lost.

If you enter a new DJAR definition name, a message is displayed, indicating that you have changed the DJAR name, and prompting you to press enter to continue. If any other key is pressed as a response to this message, the DJAR definition name reverts to the current name.
- This action is equivalent to restarting CREA after quitting.
- It allows you to change the DJAR definition name if you realize that the DJAR definition that you have started to work with is not the one you intended.
- Changing the DJAR name results in the loss of all of the changes you have made to tranID & bean/method associations since this screen was first displayed.
- The screen is redisplayed, showing the new DJAR definition's beans and methods, along with any known tranIDs.
- A **+** symbol is displayed in the line above the top method, or the line below the bottom method if more data is available.
- You can scroll up and down the contents of the JAR file using the **PF7** and **PF8** keys, and you can hide the methods associated with each bean by pressing **PF9**. When **PF9** is pressed to hide the methods, the **PF9** legend will change to read **Show Methods**, and this is reversed when it is pressed again.
- Keys legends **PF7**, **PF8** and **PF9** only appear when they are relevant. If all the information appears on one screen, **PF7** and **PF8** are not displayed. At the top or bottom of the information, only one of them is shown. If there are no methods associated with the beans, being displayed, **PF9** is not shown.
- When you scroll down, the line displayed at the bottom of the previous screen, becomes the top line of the next screen, and when you scroll up, the previous top line becomes the bottom line of the next screen.
- Pressing a function key or enter commits any changes entered on the current displayed window.
- The clear key cancels all changes made to the current screen displayed window.

When you have finished associating tranIDs with beans and methods, you can press **PF5** to create the REQUESTMODEL definitions reflecting the tranID & bean/method relationships. This takes you to the REQUESTMODEL Specification screen (Figure 145 on page 607).

Setting tranIDs for different kinds of remote method calls

You might consider three kinds of remote method calls that may be invoked for a given enterprise bean.

- The business methods, as defined in the component interface class
- The infrastructure methods, inherited from the component interface super class (javax.ejb.EJBObject)
- The methods on the home interface, (these are remotely callable methods)

The primary purpose of CREA is to define REQUESTMODELs for the business methods, but CREA treats all three alike, so if, for example, you decide that a particular method on a bean is to run under a different transaction ID from the others, or that all the remote methods for a given bean are to run under the same (but not the default) default transaction ID, you can use CREA to arrange this.

In Figure 144 on page 607, a DJAR called HelloWorldEJB, which contains a single enterprise bean called 'HelloWorld', is being processed. This bean has one business method, called 'hello', defined on the component interface. An existing REQUESTMODEL has been defined in the region which maps the 'hello' method of this bean to the 'WIBB' transaction ID. All other methods run under the default CIRP transaction ID.

Note: WIBB and the CIRP above it appear in white, the CIRPs below WIBB appear in green.

The line which contains the bean name ('HelloWorld') can be used to set a default transaction ID under which all methods on this bean will be invoked (including methods on the home interface). Below this are listed in alphabetical order the business methods of the bean. In this case, there is only one business method. Below the business methods are listed the five methods inherited from the component interface's super class (javax.ejb.EJBObject) - these same methods will be listed for all enterprise beans. Methods on the Home interface are not listed. If you want all the remote methods, including those on the home interface, to run under a specific transaction ID then supply new default against the line which lists the bean name. This causes a REQUESTMODEL to be created with an interface type of 'both'. All other REQUESTMODELs are generated with an interface type of ‘remote' and are only applied to the component interface.

If multiple overloaded method names (methods with the same name but different parameters) exist for a particular bean then a method name that includes wildcard characters will also be shown among the list of business methods. This is not illustrated in Figure 144 on page 607. You can set a transaction ID against this method name (that includes wildcard characters) in order to set the preferred transaction ID for all the overloaded instances of that method.

In all cases, the most specific REQUESTMODEL defined is the one honoured at run time. So, a REQUESTMODEL set against a specific method will always be used in preference over one set against a set of overloaded methods, and this overloaded REQUESTMODEL will be used in preference to the default set against the entire bean.

```
CREA  REQUESTMODEL Definition Transaction           APPLID

Associate a transaction ID with a bean or bean method.

DJAR name: HelloWorldEJB                       TranID filter: ____

TranID  Bean/Method/Parameters


CIRP     HelloWorld
WIBB        hello
                java.lang.string
CIRP >      getEJBHome (inherited from EJBObject)
CIRP >      getHandle (inherited from EJBObject)
CIRP >      getPrimaryKey (inherited from EJBObject)
CIRP >      isIdentical (inherited from EJBObject)
               javax.ejb.EJBObject
CIRP >      remove (inherited from EJBObject)



<Message Line>
 PF1=Help  3=Quit    5=Create ReqMods                      9=Hide Methods
```

Figure 144. CREA transaction: TranID Specification screen

REQUESTMODEL Specification

The REQUESTMODEL Specification screen (Figure 145) shows you, one by one, each of the REQUESTMODEL definitions that is to be created.

```
CREA CICS REQUESTMODEL Definition Transaction           APPLID

Create a REQUESTMODEL definition

Group: _____   Define to CSD: N   Install to CICS: N   Replace Dups: N

REQUESTMODEL name CREA_____              CORBASERVER EJB1

TranID CIRP

Bean CICSSample_____________________________________________
_____________________________________________________________
_____________________________________________________________
_____________________________________________________________

Operation *_________________________________________________
_____________________________________________________________
_____________________________________________________________
_____________________________________________________________


<Message Line>
 PF1=Help   3=Quit     8=Skip ReqMod     ENTER=Create ReqMod
```

Figure 145. CREA transaction: REQUESTMODEL Specification screen

For each REQUESTMODEL definition, the following information and input fields are shown shown:

Group: _____
If you select **Define to CSD** by entering a **Y**, .The **Group** field must contain

the name of the group that is to hold the definition. The value given for **Group** becomes the default for each subsequent REQUESTMODEL definition, but can be changed again at any point.

Define to CSD: _
Indicate where you want to create the REQUESTMODEL definition by entering a **Y** in the **Define to CSD:** field, or the **Install to CICS:** field, or both.

If you select **Define to CSD:**, the **Group** field must contain the name of the group that is to hold the definition. The value given becomes the default for each subsequent REQUESTMODEL definition, but can be changed again at any point.

Install to CICS: _
Indicate where you want to create the REQUESTMODEL definition by entering a **Y** in the **Install to CICS:** field, or the **Define to CSD:** field, or both.

Replace Dups: _
To overwrite existing definitions (either already installed in CICS, or already defined in a group on the CSD), enter a **Y** in the **Replace Dups:** field. If this field is **N**, the existence of duplicate definitions is reported as a warning. When the warning is issued, you can choose to replace this duplicate definition by pressing **PF9**, or skip to the next definition with **PF8**.

Initially a default of **N** is used. When you change this, the change become the new default for subsequent REQUESTMODEL definitions.

REQUESTMODEL name: _____
For each REQUESTMODEL definition, a generated name is shown. As each is presented, you can:

Accept
this name and move on to the next REQUESTMODEL definition. For details about the naming of REQUESTMODEL definitions, see "CREA generated REQUESTMODEL names" on page 609.

To create the definition, press enter. Successful creation of the REQUESTMODEL definition causes the next to be displayed.

Also displayed is a message indicating the name of the REQUESTMODEL definition that you have just completed and where it was defined (installed into CICS, written to the CSD, or both).

When the last REQUESTMODEL definition has been processed, the Results screen is displayed with no message indicating creation (as that information is available as part of the Results information).

If the creation of a REQUESTMODEL definition fails for any reason, a message is returned to the user, with a redisplay of the screen that requested the creation of the failing REQUESTMODEL definition. This gives you the opportunity to correct any problems with the group or the name.

When you ask for both **Define to CSD:** and **Install to CICS:**, the creation of the REQUESTMODEL definition into CICS is attempted first. If the creation does not work, the definition is not written to the CSD. If the creation of the REQUESTMODEL definition into CICS works, but the definition to the CSD fails, and you make changes to the name or tranID, the REQUESTMODEL installed in CICS is altered (or deleted and recreated) as appropriate.

Override
the generated name, then accept the new name and move on. For

details about the naming of REQUESTMODEL definitions, see "CREA generated REQUESTMODEL names."

To create the definition, press enter. Successful creation of the REQUESTMODEL definition causes the next to be displayed. The remainder of **Override** processing is the same as for **Accept**.

Skip this REQUESTMODEL definition and move to the next.

If you do not want to create the REQUESTMODEL definition displayed, press **PF8** to move to the next definition, or to the Results screen if there are no more. If you choose to skip the REQUESTMODEL definition after an error has occurred, and the REQUESTMODEL definition was installed into CICS, but could not be defined to the CSD, the REQUESTMODEL definition is removed.

Quit at any point.

To quit, press **PF3**

CORBASERVER: ____
The CORBASERVER, field is not available for change.

TranID: ____
The TranID field is not available for change.

Bean: ____
The Bean field is not available for change.

Operation: ____
The Operation field is not available for change.

CREA generated REQUESTMODEL names

CREA generates and displays a name for each REQUESTMODEL definition that is required. That generated name is produced as follows:

- The initial name is generated by suffixing a count to **CREA**, producing *CREA1*, *CREA2*, and so on, to *CREA9999*.
- When the count becomes large enough that the prefix and suffix together exceed eight characters, the prefix is truncated to allow the process to continue, so *CREA9999* is followed by *CRE10000* thru *CRE99999*, then *CR100000* thru *CR999999* and so on. If , following this rule, the name reaches *99999999*, then incrementing ceases, and you are presented with the same REQUESTMODEL name again.
- You can overtype this name with one of your choice.
- If you do, you change the basis for the next REQUESTMODEL default name. If your overtyped string includes a trailing count, this is used as a starting point, with 1 being added to this count for each new REQUESTMODEL.

  For example, if you overtype *CREA1* with:
  - *SHEEP*, the following REQUESTMODEL name displayed is *SHEEP1*,
  - *RHINO26*, the following REQUESTMODEL name displayed is *RHINO27*,
  - *GIRAFFE9*, the following REQUESTMODEL name displayed is *GIRAFF10*.

  The same rules about truncation of the prefix apply when the succession of names is based on a string that you previously supplied by overtyping. This is true both when incrementing is on the point of taking you beyond the eight character maximum, and when you supply an over-long prefix by overtyping, for example, if you overtype with *ELEPHANT*, the following REQUESTMODEL name displayed would be *ELEPHAN1*.

Results of using CREA

When you have finished creating REQUESTMODEL definitions, the REQUESTMODEL results screen displays, summarizing what you have done. It lists the REQUESTMODEL definitions that were installed into CICS and the REQUESTMODEL definitions that were written to the CSD. If more than one group was used to hold definitions, each group is listed with the REQUESTMODEL definitions it contains.

```
   CREA CICS REQUESTMODEL Definition Transaction            APPLID

REQUESTMODEL definition is complete.

REQUESTMODELs were processed for DJAR SampleEJB

The following REQUESTMODELs were installed into CICS:
RM1, RM2, RM3, RM4, RM5

The following REQUESTMODELs were written to the CSD in group YOURNAME:
RM2, RM5

The following REQUESTMODELs match one or more beans in the DJAR, but either
do not match any methods on that bean, or have been superseded by a newly
created REQUESTMODEL.  These REQUESTMODELs are no longer needed and may
be deleted:
RM7, RM8

The following REQUESTMODELs contain a bean name that matches one or more of
the beans in the DJAR because wildcard characters were used in the bean name.
These REQUESTMODELs are no longer needed to match the beans in this DJAR and
may be deleted if other beans in other DJARs no longer require them:
RM10, RM11

The following TRANIDs were used, but are not currently installed:
TRN1, TRN2, TRN3, TRN4

<Message Line>
 PF1=Help   3=Quit                          12=New DJAR
```

Figure 146. CREA transaction: REQUESTMODEL results screen

The following messages appear on this screen:

1. `REQUESTMODEL definition is complete.` always appears first.
2. `REQUESTMODELs were processed for DJAR` *`SampleEJB`*`.` always appears next. The DJAR name shown is the one you specified at the start of this use of CREA.
3. The next message is either `The following REQUESTMODELs were installed into CICS:` *`list of REQUESTMODELs`*, or `No REQUESTMODELs were installed into CICS.`
4. The next message is either `The following REQUESTMODELs were written to the CSD in group` *`Yourname`*`:` *`list of REQUESTMODELs`*, or `No REQUESTMODELs were written to the CSD in any groups.` The group name, shown here as *yourname* is whatever you supplied previously, on the REQUESTMODEL Specification panel.
5. The following messages appear only if appropriate:
   - `The following REQUESTMODELs were found to be related to a bean in the selected DJAR, but not matched to a bean or method: RM7, RM8`

     Such REQUESTMODEL definitions might be redundant and could potentially be removed from the CICS system (and from the CSD if applicable).
   - `The following REQUESTMODELs were found to be related to a bean in the selected DJAR, but are no longer use by the selected DJAR: RM10, RM11`

Such REQUESTMODEL definitions are no longer required by the bean and could potentially be removed from the CICS system (and from the CSD if applicable).

- `The following TRANIDs were used, but are not currently installed: TRN1, TRN2, TRN3, TRN4`

In these messages, RM7, RM8, RM10, RM11 and TRN1, TRN2, TRN3, TRN4 are examples of names for REQUESTMODELs and TRANIDs. For each of the three messages above, there is no message to indicate the absence of that condition.

If the information extends beyond one screen, use PF7 and PF8 to scroll up and down the information presented. To work with another DJAR definition, restart CREA by pressing F12 to return to the initial, (or DJAR Specification), screen (Figure 142 on page 603).

CREC

The CREC transaction, allows you to use those functions of CREA, which display information, but you cannot update either the CSD file or the running CICS system. By default, users who are authorized to use CEDC can use CREC.

Using CREC

Use CREC in the same way that you use CREA. Because CREC only displays information, and changes nothing there are two significant differences to note:

- The TranID Specification screen (Figure 143 on page 604) displays the tranID fields next to the beans and methods in a read only state. The F5 option is unavailable and does not appear.
- The REQUESTMODEL Specification screen (Figure 145 on page 607), and the REQUESTMODEL results screen (Figure 146 on page 610) are not available.

Chapter 30. CRTE - remote transactions

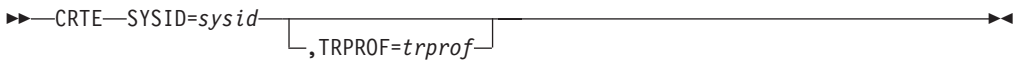
Use CRTE to run transactions that are on a remote system.

For transaction routing, you create resource definitions that define the transaction as remote and name the system on which it is to be processed. Entering the transaction code then causes CICS to route the requests to the named system.

The CRTE transaction provides you with another way of running a transaction on a remote system. Use CRTE, rather than creating special resource definitions, for infrequently used transactions or when using transactions such as CEMT, which are on all systems.

You must use CRTE when EDF is used to test a transaction running on a remote system, although EDF is not supported over an IPIC connection. However, because the session has to be established and canceled explicitly, additional sign-on operations might be required. You can start CRTE from a 3270 terminal, but not on LU6.1 links.

CRTE



where:

sysid
: Is the name of the system that will process the transaction. This *sysid* is the same as one of the following identifiers:
  - For MRO and APPC connections, the 4-character identifier of the remote system, as specified in the name of its CONNECTION resource definition.
  - For IPIC connections, the first 4 characters of an IPCONN that is in service and acquired.

trprof
: Is the 8-character name of the profile that is used for the session that carries the intersystem flows, if that session uses z/OS Communications Server. If z/OS Communications Server is used and this parameter is not entered, the default profile DFHCICSS is used.

The routing transaction verifies that the specified remote system is known and is available. If it is, a message is displayed confirming that a routing session to the required system has been started.

When you have received this message, you can clear the screen and enter the transaction identifier for the transaction that is run on the remote system. In fact, you can use the terminal as if it were connected directly to the remote system, although you might be asked to sign on to the remote system before being allowed to run any transactions. The transactions that you can start include pseudoconversational transactions, BMS paging transactions, and CRTE itself. However, you cannot use transactions CEOS and CEOT with CRTE, and you cannot use PA or PF keys to start transactions under CRTE.

© Copyright IBM Corp. 1982, 2011

You end a routing session by entering CANCEL. If you have signed on to the remote system, the CANCEL request automatically signs you off.

When a routing session ends, you get the message:
`ROUTING SESSION TO SYSTEM system-id HAS BEEN TERMINATED`

Note:

1. If you use CRTE to route transactions through more than one system, to end the routing session you must press CANCEL the same number of times as you entered CRTE.
2. While a terminal is in a routing session with another system, that is, during the period between entering CRTE and CANCEL, the terminal cannot receive messages delivered by automatic transaction initiation (ATI).
3. The message cannot be delivered immediately if a transaction meets one of these criteria:
   - Runs on a remote system using CRTE
   - Builds a message using BMS paging
   - Sends the message using a **BMS SEND PAGE** command
   - Does not specify either RETAIN® or RELEASE

   It is delivered when the routing session ends, but only if the terminal was in TRANSCEIVE status when the routing session started. Alternatively, you can retrieve the message, while still routed, using BMS paging commands.
4. These restrictions apply to the use of CRTE with BMS chaining of messages:
   - You cannot use CRTE as a chained transaction without losing the logical message from which CRTE was to be chained.
   - You cannot use chaining in a routing session without purging the logical message currently attached to the terminal.
   - You cannot route transactions back to any system in the chain.

Chapter 31. CSFE - terminal and system test

Use CSFE to diagnose terminal problems and software problems. This transaction is mainly intended to be used by system programmers and IBM field engineers.

Use this transaction to:

- Test your terminal
- Activate the system spooling interface trace
- Activate the terminal builder trace
- Activate any of the following: storage freeze, the storage violation trap, and the global trap/trace program exit routine.

For each of these activities, successful completion of the CSFE transaction gives you the following message:

```
DFHFE3301 Transaction complete
```

For information about the use of this transaction, see the *CICS Problem Determination Guide*.

Using CSFE terminal test

You can use CSFE to test your terminal, but only if it is one of the terminals supported by CICS.

About this task

You cannot use CSFE to test the following:

- IBM 2780 Data Transmission terminals
- IBM 3600 Finance Communication System terminals
- IBM 3614 Consumer Transaction Facility terminals
- IBM 3735 Programmable Buffered terminals
- Output-only printers (for example, printers of the IBM 3270 Information Display System).

Procedure

1. You start the transaction by entering CSFE on the command line of the screen, as follows:

   ```
   CSFE
   ```

2. Press the ENTER key. You get the following display:

   ```
   DFHFE3304  Enter PRINT for character set, END to
   terminate.  All other data will be echoed.
   ```

   END
   When you type END on the command line of the screen, the test ends.

   PRINT
   When you enter PRINT, all characters printable or capable of display at the terminal are sent to your terminal.

   Any other input is returned to your terminal (that is, it is echoed), unless it is of the form of one of the other CSFE requests.

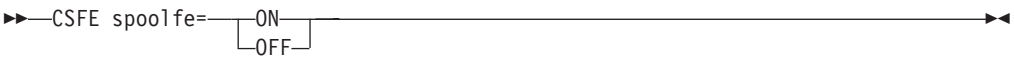
© Copyright IBM Corp. 1982, 2011

Tracing with CSFE

You can use CSFE to request additional trace and diagnostic facilities for the system spooling interface.

About this task

CSFE SPOOLFE



OFF
switches off the system spooling interface trace and the diagnostic facility.

ON
specifies that additional trace entries of system type are provided for the system spooling interface.

SPOOLFE
is the system spooling interface trace and diagnostic facility.

Terminal builder trace

You can use the CSFE transaction to activate and deactivate the terminal builder trace.

CSFE ZCQTRACE

,AUTOINSTALL
Switches tracing on for autoinstalled terminals.

,OFF
Switches tracing off for individual and autoinstalled terminals.

=termid
Switches tracing on for the terminal or connection identified by termid.

Using CSFE debug functions

About this task

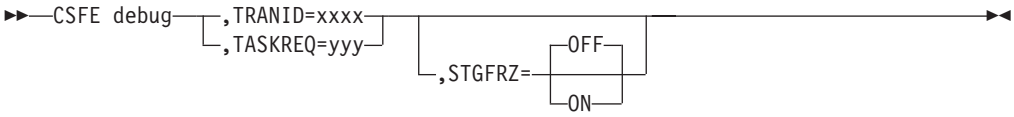
Storage freeze, the storage violation trap and the global trap/trace exit are all debugging functions that are controlled by the CSFE transaction. These functions are described in the following sections.

Storage freeze

You can use CSFE DEBUG to freeze storage that is normally freed during the processing of a transaction. The frozen storage is not freed until the end of the transaction. This freezing of storage can help you in the subsequent diagnosis of problems that might have occurred during processing.

You must code the options in the order shown in the syntax display.

CSFE DEBUG TRANID



Parameters

STGFRZ= {**OFF**|**ON**}
Specifies that you want the storage freeze function to be set on or off. If you omit this option, storage freeze is set to OFF by default. To improve efficiency, storage freeze should be switched off when not required.

TASKREQ= *yyy*
This is an alternative method of identifying the transaction. *yyy* is the function key definition on a 3270 display device which identifies the transaction for which storage freeze is to apply; for example, TASKREQ=PA1.

TRANID= *xxxx*
Identifies the transaction for which storage freeze is to apply.

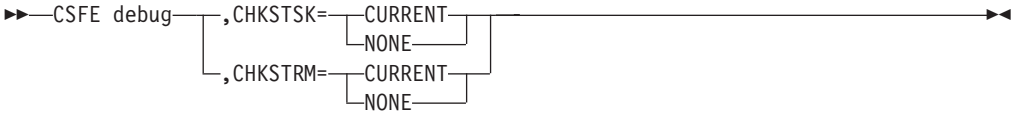
Storage violation trap

Use CSFE DEBUG to activate and deactivate the storage violation trap, which is in the trace program (DFHTRP), the exec interface program (DFHEIP), and the resource manager interface program (DFHERM).

This trap lets you verify the user storage associated with the currently active task. The storage violation trap is entered before the trace program returns to its caller after making a trace entry.

When a storage violation is detected by the trap, an exception trace entry is made, a CICS system dump is produced (if dumping has not been disabled), and the storage violation trap is switched off.

CSFE DEBUG CHKSTSK



Parameters

,CHKSTSK= {**CURRENT**|**NONE**}

CURRENT
Specifies that validity checks are to be performed on the user storage (not the CICS storage) for the current task whenever the trace program is entered. Validity checks are to be performed for the currently active task, as follows:

- Validity checking of the storage chain for the task
- Validity checking of the addresses and storage accounting areas for all storage areas on the transaction storage chain.

NONE
Deactivates the user storage checking function.

,CHKSTRM= {**CURRENT**|**NONE**}
The terminal subpools, above and below the line, are to be checked for the CURRENT task or for NONE (no tasks).

Global trap/trace program exit and global kernel exit

You can use CSFE DEBUG with the TRAP or GTRAP operand to activate and deactivate the global trap/trace program exit routine or the global kernel exit routine respectively.

These exits are intended to be used only under the guidance of IBM support personnel.

CSFE DEBUG TRAP

TRAP={**ON**|**OFF**}
The global trap/trace program exit routine is to be activated or deactivated.

GTRAP={**ON**|**OFF**}
The global kernel exit routine is to be activated or deactivated. On each domain call and return and each subroutine call and return, the kernel invokes the global kernel exit routine if it is activated.

Chapter 32. CSPG - page retrieval

Some transactions, especially those performed on screens, produce a response that is too large to be displayed or printed at one time. The whole response is held on direct access storage, and only the first portion (called a page) is displayed. You can manipulate these pages, through basic mapping support (BMS), using the CSPG page retrieval transaction.

You can use paging to view all pages of the response, in any order. All the pages produced by a transaction make up a logical message.

A page retrieval session begins when you enter the first page retrieval command, or when the first page is presented by automatic transaction initiation (ATI). During the session, CICS passes commands directly to BMS terminal paging, without following the normal transaction identification procedure.

When using BMS for IBM 3780 Communication Terminal transactions, switch off the SPACE COMPRESS EXPAND (standard feature), TRNSPCY (special feature), and the INHIBIT IRS (special feature) switches.

Pages to be manipulated with paging commands must have been created by BMS, which must be active in the system to service the commands. Paging commands enable you to:

- Retrieve a page.
- Display information about logical messages queued for your terminal.
- Chain a logical message, that is, retain the previous logical message on direct access storage while you enter a new transaction.
- Purge a logical message, that is, erase it from direct access storage; a paging session must be ended by purging the message.
- Copy a page to another terminal.

You select the above functions by entering a unique identification code for each function. This code is 1–7 characters long, and is defined by the PGRET, PGCHAIN, PGPURGE, and PGCOPY system initialization parameters.

Do not use codes that are the same as the leading characters of any transaction defined to CICS by a TRANSACTION resource definition.

Retrieve a page

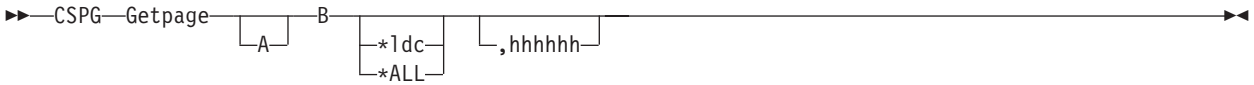
The CSPG page retrieval command allows you to retrieve a specified page. If partitions are in use, the page retrieval command refers to the partition in which it has been entered.

About this task

The syntax and attributes of the CSPG Getpage command are as follows:

© Copyright IBM Corp. 1982, 2011

CSPG GETPAGE



Getpage
This is the user-defined identification code for the retrieval transaction, and is 1–7 characters long. This code is defined by the PGRET system initialization parameter. For convenience, make the code as short as possible; for example, P/.

A You can ignore this option unless chaining is being used. It is the level, within a chain of logical messages, of the logical message to be retrieved. You must enclose it in parentheses. It can be one of the following:

n message is at level n

+n message is n levels forward

-n message is n levels backward

P previous level backward

C current level (the default)

N next level forward

L last level.

For more information about levels of messages, see "Message chaining with CSPG" on page 622.

B Defines the next operation to be performed on the logical message at the current level, or at the level specified by the A option it can be one of the following:

n page n

+n page n pages forward

-n page n pages backward

P previous page

C redisplay current page (after clearing the screen)

N next page

L last page

A for an autopaging (that is, hard copy) terminal that is temporarily in paging mode: return the terminal to autopaging mode and output all remaining pages.

This is discussed further under "Using Autopage with CSPG" on page 627.

X enter data that is not recognized as a paging command.

This is discussed further under "Single keystroke retrieval (SKR)" on page 621.

***ldc**
Is the LDC for the device to which you want to send the message. (See "Page retrieval for logical devices.")

***ALL**
A special case of *ldc.

Hhhhhh
Is the 6-digit hexadecimal identifier of an undelivered logical message to be retrieved. You would normally determine this identifier by entering a page query command (see "Display information about logical messages" on page 622).

If you do not specify an identifier, the oldest message is retrieved. After a paging session has started, you can omit hhhhhh because all subsequent pages are from the current message.

In the following examples, P/ is the page retrieval transaction identifier:

P/7 Retrieve page 7 of this message.

P/N Retrieve next page (following the one being displayed) of this message.

P/(N)9 Retrieve page 9 of the next message (of several that have been chained).

Single keystroke retrieval (SKR)

If you use a 3270 display device under z/OS Communications Server, you can use PA or PF keys, as defined in the SKRxxxx system initialization parameter, to enter page retrieval commands, and greatly reduce the number of keystrokes necessary in a page retrieval session.

After the session has ended, the PA and PF keys can again be used for initiating transactions in the usual way. The *CICS System Definition Guide* gives further information about this topic.

Page retrieval for logical devices

Your application program can send messages to a communication system such as the IBM 3600. Your application program can also send different pages to each of the logical devices within the system. These pages must be grouped as a single logical message, held in storage.

Each part of a logical message is accompanied by a logical device code (LDC) that identifies its destination. When retrieving the pages for a logical device, you must specify the LDC for that device. However, you do not need to do this if the user-written program in the logical device controller provides a default LDC mnemonic.

If you specify the page number and *ldc, delivery begins with the specified page for that logical device. If the logical device is in AUTOPAGE status, all pages for that logical device, starting with the requested page, are written. If the logical device is in PAGE status, only the specified page is written.

If you omit *ldc, and the application program in the logical device does not provide a default, BMS selects the first page destined for a logical device that is currently in PAGE status. The application program in the logical unit should provide a default LDC, because pages may be delivered to logical devices other than those you originally specified.

In either case, if your request is the first request for a page from a logical message, all pages (within the logical message) destined for logical devices currently in AUTOPAGE status are delivered after the first or specified page is delivered to a device in PAGE status.

The getpageA command supports an *ldc specification for nonrouted messages created for a 3601 logical unit. The various forms of getpageA are interpreted as follows:

- getpageA*ALL-resets to autopaging and delivers all pages for the logical devices that have autopage specified in the terminal definition, but that were temporarily placed in page status by the CICS application programmer.
- getpageA*ldc-resets the specified logical device to autopage and delivers the pages for that *ldc. The command is similar to getpageA*ALL, except that it operates only for the specified logical device.
- getpageA-resets to autopaging and delivers the remaining pages for the logical device for which pages are currently being displayed.

Page retrieval and partitions

The page retrieval commands, including any PA and PF keys used in single keystroke retrieval, refer to the partition in which they are entered. The partition in which the last paging command is entered is the *current partition*. A paging command for a partitioned message entered on a nonpartitioned screen (that is, after CLEAR) refers to the current partition.

After a transition from base (unpartitioned) state to partitioned state, or a change of partition set (that is, the initial page display, a page display following CLEAR, or following a change in the level of page chaining), the *current page* is displayed in each partition. The current page is initially the first page for that partition. It is updated by page retrieval commands.

Display information about logical messages

The **getpageQ** command displays the message identifiers and titles of logical messages queued for this terminal.

Getpage
The user-defined identification code for the retrieval transaction, and is 1–7 characters long. The PGRET system initialization parameter defines this code.

Q (Query). The identifiers of all logical messages destined for this terminal are displayed. If the message is security protected (that is, if it has been routed to a specific operator ID or operator class), its identifier is displayed only if the operator ID or class for the operator signed on matches that in the message. The identifier consists of a 6-digit hexadecimal number, a 2-character temporary-storage prefix, and optionally, a title. The 6-digit hexadecimal number can be used as the hhhhhh option in the page retrieval command (see "Retrieve a page" on page 619) to retrieve pages for this display.

Message chaining with CSPG

Usually, when you have viewed the output pages from a transaction, there is no need to retain the pages. However, you might want to initiate a new transaction and still retain the old pages, so that you can view them again after you have displayed output from the new transaction. This is called *message chaining* and can be done using the following command.

CSPG MESSCHNYYYY

Messchn
The installation-defined message-chaining transaction identification code (for example, C/), as defined in the PGCHAIN system initialization parameter.

Yyyy
The identifier of a transaction that produces data to be added to the message chain.

Parameters
The parameters, if any, required by transaction yyyy.

If the new transaction uses terminal control, or BMS in TERMINAL mode, its output is displayed immediately. When the new transaction terminates, anything else you enter is interpreted as a paging command for the first message. For example:

TRAN The transaction invokes an application program that produces multiple pages of output. The first page is displayed.

P/2 The second page of message is displayed.

C/CEMT INQ TIM
CEMT data is displayed.

P/2 The second page of message is redisplayed.

If the new transaction uses BMS in PAGING mode, and RETAIN or RELEASE is specified in the SEND PAGE command in the application program, the output is chained to existing output, and a new level of chaining is created.

If RELEASE is specified, this chaining process can be repeated indefinitely. However, chaining requests must always be made while viewing pages at the highest level of chaining. Otherwise, messages at levels higher than the current level is automatically purged (if OPERPURGE has not been specified in the application program SEND PAGE command), or you are asked to purge them. If RETAIN is specified, no further chaining (in either TERMINAL or PAGING mode) can be performed until the second logical message has been terminated. Further chaining requests are rejected.

If neither RETAIN nor RELEASE is specified, the output from the second transaction is queued as a new message for the terminal, and is not received until all earlier messages have been terminated. It is thus not chained to the current message.

If you enter data that is not a paging command during a message chaining session, all pages on levels higher than the current level are deleted, if no message in the chain has been built using OPERPURGE=YES in the SEND PAGE command.

An example of the use of message chaining under BMS in PAGING mode is a search of a personnel file for people who speak French. Your initial inquiry might turn up three names. Without terminating the page containing the names, you can examine the records of each of the three people named. The chaining transaction could be used to read a record, without erasing the three names.

This would create output at the second level of chaining (the output from the original transaction being first-level output). When you finish viewing a person's record, you enter a paging command to terminate the current level, see “CSPG message termination” on page 625. The message displaying the person's record would be purged, and the original page, containing the three names, redisplayed. Assuming that the message chaining transaction identifier is C/, enter:

```
LANG DEPT=J34,FRENCH
```

Receive:

```
BERESFORD, J.A. 1043
LEVY, A.R.      7121
WALTON, J.      8143
```

To find out more about Beresford, enter:

```
C/PREC 1043
```

This finds the personnel record for employee number 1043. PREC is the identifier of a user-written transaction for examining personnel records. Receive:

```
NAME - BERESFORD, JERRY A.
EMPLOYEE NO - 1043
(and so on)
```

To go back to the list of names, enter:

```
T/C
```

T/ is the user-defined transaction identifier that terminates the message at the current level of chaining (see “CSPG message termination” on page 625).

Receive:

```
BERESFORD, J.A.  1043
LEVY, A.R.       7121
WALTON, J.       8143
```

You can now examine Levy's record.

You can also chain a series of messages together and use paging commands to view the series simultaneously. Using the previous example, you can have the summary page as your base message (first level) and each of the personnel records as subsequent chain levels. Assume you are currently displaying the summary:

Enter:

```
C/PREC 1043
```

Receive:

```
NAME - BERESFORD...
```

Enter:

```
C/PREC 7121
```

Receive:

```
NAME - LEVY...
```

Enter:

```
C/PREC 8143
```

Receive:

```
NAME - WALTON...
```

There are now four levels of logical messages at the terminal. You can now page back and forth between the personnel records. Enter:

```
P/(2)1
```

Receive:

```
NAME - BERESFORD...
```

Enter:

```
P/(L)1
```

Receive:

```
NAME - WALTON...
```

You can chain a new transaction only while you are viewing the last logical level. If you have four levels (as in the above example), view the summary page (level one) and then enter C/PREC 3642; the new transaction becomes level two and all other chained messages below that are erased (assuming that they have been built using a SEND PAGE command with the OPERPURGE option omitted).

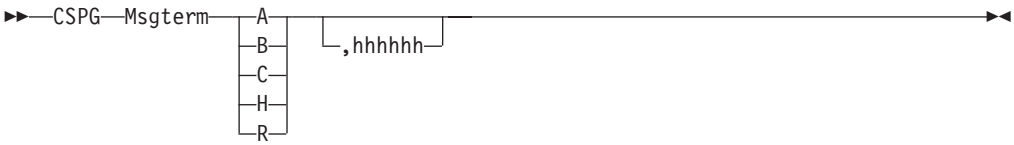
Messages involving different partition layouts (different partition sets, including unpartitioned) may be chained. The current page in each partition is redisplayed on a change of message level.

CSPG message termination

You can erase a logical message using the message terminate command.

If the logical message is security protected, it is terminated only if your identification or class matches that in the message. After an erase, the last message still in the chain is displayed. The format of the message terminate command is as follows:

CSPG MSGTERM



Msgterm
A user-defined transaction identifier (1–7 characters) for the message termination transaction (for example T/) as defined in the PGPURGE system initialization parameter.

A Terminates all logical messages destined for and being displayed on that terminal (including routed messages not yet displayed). The following message is received when termination has been completed:

```
DFHTP4122 Requested purge completed successfully
```

B Terminates the logical message being displayed on that terminal and all logical messages chained to it. This does not include routed messages. Whenever you have finished with all the output from a transaction, you should terminate the

base message by entering a command of the form msgtermB. If the message has been built using a SEND PAGE command with the OPERPURGE option omitted, you can terminate the paging session by entering data other than a paging command.

C Terminates the current (level) logical message and any messages chained below it.

H Terminates all logical messages chained to the base logical message being displayed on that terminal. The page of the base message that was most recently displayed is redisplayed at the terminal.

R Terminates all logical messages queued for immediate delivery (routed) to the terminal. The page that was being displayed at the terminal is redisplayed.

Hhhhhh
The 6-digit hexadecimal identifier of a specific, undelivered logical message to be erased. (This identifier would have been determined using a page query command.) If the message specified is security-protected, the operator should have a security code that matches the one in the message.

If a message identifier (hhhhhh) is not specified in the termination request and there is no message currently displayed, the oldest message is automatically retrieved.

Copying pages with CSPG

You can send a copy of the page being displayed, as it appeared when initially transmitted, to another terminal (normally hardcopy).

About this task

You do this as follows:

Procedure

Enter the page-copy transaction identifier and the identifier of the terminal that is to receive the copy, as follows:

```
msgcopytttt
```

where:

Msgcopy
The copy transaction identifier (1–7 characters), for example D/, as defined in the **PGCOPY** system initialization parameter.

Tttt
The identifier of the terminal to which the page is to be copied.

Results

You get the message:

```
DFHTP4124 Page copied from terminal tttt
        (message number msgno)
```

Field and character attributes in the copied page that are not supported by the target terminal are ignored. The copied page is reformatted if it does not fit on the target terminal. Field and character attributes are discarded in the reformatting. If partitions are in use, the page copy command copies the displayed page for the

partition in which the copy command is entered.

Using Autopage with CSPG

A hardcopy terminal that is not defined as a paging terminal (one that displays pages when you request them) is an autopaging terminal (one that displays all pages without your intervention).

About this task

An autopaging terminal may be temporarily designated as a paging terminal by the user's application program using the NOAUTOPAGE option of the SEND PAGE command. If this is done at the time the first page is to be displayed on the terminal, the terminal is temporarily treated as a paging terminal for the duration of the message.

This condition exists until either of the following occurs:

1. You enter P/A (reset terminal to autopaging status), indicating that all subsequent pages are to be displayed without further intervention.
2. You enter T/C (terminate current message). All pages from that logical message are purged, and the terminal is again put into autopaging status.

P/ is the paging transaction identifier, and T/ is the terminate transaction identifier.

Displaying routed pages with CSPG

When a terminal to which you have directed a message is in TRANSCEIVE status, and is not involved with a transaction, and is not displaying pages, the first page of that message is displayed. However, if the terminal is in TRANSACTION status, you can display the first page only of a waiting message by entering CSPG, CSPG P/1, or just P/1.

About this task

Here, you are using P/ as a paging command. If no page is waiting, you get the following message:

```
DFHTP4104 A paging request has been received but
    there are no pages for display
```

You can use the message-switching transaction to direct output to a terminal or terminals other than your own. For more information about message switching, see Chapter 27, “CMSG - message switching,” on page 573.

Chapter 33. CWTO - write to console operator

Use CWTO to send messages to the console operator.

Start the transaction by entering CWTO, optionally followed by the text of your message, on the command line of the screen, as follows:
Press the ENTER key. If the transaction runs successfully, your message is sent to

CWTO

the console, and you get the normal completion response:

```
MESSAGE HAS BEEN SENT
```

If you enter CWTO with no message text, you are prompted for your message as follows:

```
ENTER MESSAGE
```

You can cancel the CWTO transaction by entering CANCEL as the last 6 characters of your message. The transaction is terminated and you get the following response:

```
TERMINATED BY OPERATOR
```

The following are some general points about the format of your messages:

- If your message exceeds the line length of the console, it is divided between words into more than one line. Each part of a divided message is prefixed by your identifier and by that of the originating terminal.
- New-line (NL) characters within your message are kept, although any leading NL character is deleted.
- You can include blank lines at the top of your transmitted message by pressing ENTER (or using carriage returns) the required number of times.
- You can enter any number of backspace characters for deletion. If you enter only blank characters, or if you delete all of the data by entering too many deletion characters, you get the following message:

  ```
  NO INPUT - REENTER
  ```

© Copyright IBM Corp. 1982, 2011

Chapter 34. DSNC - CICS DB2 transaction DSNC

Use DSNC to enter DB2 commands and carry out a range of other CICS DB2 tasks.

The DSNC transaction can be used to perform the following:

- Enter DB2 commands from a CICS terminal.
- Cause threads to be terminated when they are released (DSNC DISCONNECT).
- Display information about transactions using the CICS DB2 interface, and display statistics (DSNC DISPLAY).
- Modify the unsolicited message destinations, and modify the number of active threads used by a DB2ENTRY, the pool, or for commands (DSNC MODIFY).
- Shut down the CICS DB2 interface (DSNC STOP).
- Start the CICS DB2 interface (DSNC STRT).

For more information about the CICS DB2 transaction, DSNC, see the *CICS DB2 Guide*.

© Copyright IBM Corp. 1982, 2011

Appendix. List of CICS transactions

These transactions are supplied with CICS, except for those that are part of CICS sample programs.

The transactions that have operator interfaces are marked by an asterisk (*). The remainder therefore have no operator interface.

Transaction is the transaction identifier. **Program** indicates the program, if there is one, that is specified in the supplied transaction definition. **CSD group** indicates the supplied resource-definition group that contains the transaction definition. For general information about resource definition and the CICS groups, see the *CICS Resource Definition Guide*.

Table 10. CICS transactions supplied by IBM

Transaction	Program	CSD group	Security category	Description
CADP*	DFHDPLU	DFHDP	2	Application debugging profile manager
CATA	DFHZATA	DFHSPI	1	Defines autoinstall automatic terminal
CATD	DFHZATD	DFHSPI	1	Deletes autoinstall terminal
CATR	DFHZATR	DFHSPI	3	Deletes autoinstall restart terminal
CATS	DFHZATS			
CAUT	DFHSTSP			Automatic System Statistics Program
CBAM*	DFHECBAM	DFHOPER	2	BTS objects browser
CBRC	DFHBRCP	DFHDLI		
CCIN	DFHZCN1	DFHCLNT	3	CICS Client
CCMF	DFHCCMF			Monitoring ATI Program
CCRL*	DFHSOCRL	DFHOPER	2	CICS certificate revocation list transaction
CDBC*	DFHDBME	DFHDBCTL	2	DBCTL interface menu transaction
CDBD	DFHDBDI	DFHDBCTL	1	DBCTL disable function
CDBF	DFHD2CM3	DFHDB2	1	CICS DB2 attachment facility shutdown force transaction
CDBI*	DFHDBIQ	DFHDBCTL	2	DBCTL interface inquiry transaction
CDBM*	DFHDBMP	DFHDBCTL	2	DBCTL operator transaction
CDBN	DFHDBCON	DFHDBCTL	Exempt (See note)	DBCTL issue commands
CDBO	DFHDBCT	DFHDBCTL	1	DBCTL control function
CDBQ	DFHD2CM2	DFHDB2	1	CICS DB2 attachment facility shutdown quiesce transaction

© Copyright IBM Corp. 1982, 2011

Table 10. CICS transactions supplied by IBM (continued)

Transaction	Program	CSD group	Security category	Description
CDBT	DFHDBDSC	DFHDBCTL	2	DBCTL interface disconnection transaction
CDFS	DFHDFST	DFHISC	2	Dynamic starts with interval
CDST	DFHMIRS		2	CICS Dynamic routing of non-term starts
CDTS	DFHZATS	DFHSPI	1	Provides remote single delete transaction
CEBR*	DFHEDFBR	DFHEDF	2	Browse temporary storage
CEBT*	-	-		Master terminal, alternate CICS
CECI*	DFHECIP	DFHINTER	2	Command level interpreter
CECS*	DFHECSP	DFHINTER	2	Command level interpreter
CEDA*	DFHEDAP	DFHSPI	2	Resource definition online - full (RDO)
CEDB*	DFHEDAP	DFHSPI	2	Resource definition online - restricted (RDO)
CEDC*	DFHEDAP	DFHSPI	2	Views resource definition online (RDO)
CEDF*	DFHEDFP	DFHEDF	2	Execution diagnostic facility
CEDX*	DFHEDFP	DFHEDF	2	Execution diagnostic facility for non-terminal tasks
CEGN	DFHCEGN	DFHSIGN	3	Schedules good night transaction
CEHP	DFHCHS	DFHISC	2	LU type 2 mirror transaction (obsolete)
CEHS	DFHCHS	DFHISC	2	LU type 2 mirror transaction (obsolete)
CEJR	DFHEJITL	DFHSTAND	3	Corbaserver resolution
CEKL*			Exempt (See note)	Master terminal for emergency use
CEMN	DFHCEMNA	DFHOPER	2	CICS monitoring facility transaction
CEMS	DFHEMSP			Spooling Master Terminal Transaction
CEMT*	DFHEMTP	DFHOPER	2	Master terminal
CEOS	DFHEOSP			Spooling Terminal Operator Transaction
CEOT*	DFHEOTP	DFHOPER	2	Terminal status
CEPD	DFHEPDS	-	1	Event processing dispatcher
CEPF	DFHECDF	-	1	Event processing deferred filtering task
CEPH	DFHECEAH	-	2	HTTP EP adapter for event processing
CEPM	DFHEPSY	-	1	Event processing queue manager

Table 10. CICS transactions supplied by IBM (continued)

Transaction	Program	CSD group	Security category	Description
CEPQ	DFHECEAM	DFHEP	2	WebSphere MQ EP adapter for event processing
CEPT	DFHECEAT	DFHEP	2	TSQ EP adapter for event processing
CESC*	DFHCESC	DFHSIGN	1	Processes timeout and signoff for idle terminals
CESD	DFHCESD	DFHSDAP	2	Shutdown assist
CESF*	DFHSFP	DFHSIGN	3	Signs off terminal user
CESL*	DFHSNP	DFHSIGN	3	Signs on terminal user with a password or password phrase
CESN*	DFHSNP	DFHSIGN	3	Signs on terminal user with a password
CEST*	DFHESTP	DFHOPER	2	Supervisory terminal
CETR*	DFHCETRA	DFHOPER	2	Inquire and set trace options
CEX2	DFHD2EX2	DFHDB2	1	CICS DB2 protected thread purge mechanism and other CICS DB2 services.
CFCL	DFHFCDL	-	1	CFDT Load
CFOR	DFHFCQT	-	1	RLS offsite recovery
CFQR	DFHFCQT	-	1	RLS quiesce receive
CFQS	DFHFCQT	-	1	RLS quiesce send
CFTI				Flowmark for MVS
CFTL	DFHDTLX	-	1	Shared DT load
CFTM				Boeblingen Banking Project
CFTS	DFHZATS	DFHSPI	1	Provides remote mass flag transaction
CFUP				Boeblingen Banking Project
CGRP	DFHZCGRP	-	1	Provides z/OS Communications Server (formally VTAM) persistent sessions (See note)
CHLP	DFHCMAC	DFHCMAC		Alias for CMAC
CIDP*	DFHDPIN	DFHDP	2	Inactivate debugging profiles utility
CIEP	DFHIEP	DFHIPECI	3	ECI for TCP/IP listener
CIND*	DFHINDT	DFHINDT	2	CICS indoubt testing tool
CINS				
CIOD	DFHIIOPA	DFHIIOP	1	Default IIOP interface, started by CIOR
CIOF	DFHIIOPA	DFHIIOP	1	CICS Generic factory, started by CIOR
CIOR	DFHIIOP	DFHIIOP	1	CICS IIOP interface, started by SO_Domain

Table 10. CICS transactions supplied by IBM (continued)

Transaction	Program	CSD group	Security category	Description
CIRB				SQL/DS Transaction ID
CIRD				SQL/DS Transaction id
CIRP	DFHIIRP	DFHIIOP	2	Default CICS IIOP request processor transaction
CIRR	DFHIIRRS	DFHIIOP	1	Default CICS IIOP request receiver
CIS4	DFHISCIS4	DFHISCIP	1	IPIC External Security Interface (ESI) transaction
CISB	DFHISCOP	DFHISCIP	1	IPIC release IPCONN on the server side of a connection (BIS processing)
CISC	DFHISCOP	DFHISCIP	1	IPIC acquire IPCONN on the client side of a connection
CISD	DFHISCOP	DFHISCIP	1	IPIC release IPCONN on the client side of a connection
CISE	DFHISCIP	DFHISCIP	1	IPIC error and message program
CISM	DFHISRSP	DFHISCIP	1	IPIC remote scheduler
CISQ	DFHISLQP	DFHISCIP	1	IPIC local queue processing
CISR	DFHISRRP	DFHISCIP	1	IPIC request/response receiver
CISS	DFHISCOP	DFHISCIP	1	IPIC acquire IPCONN on the server side of a connection
CIST	DFHISCOP	DFHISCIP	1	IPIC terminate IPCONN
CISU	DFHISREU	DFHISCIP	1	IPIC recovery transaction
CISX	DFHISREX	DFHISCIP	1	IPCONN recovery and resynchronization transaction for XA clients
CITS	DFHZATS	DFHSPI	1	Provides remote autoinstall transaction
CJGC	DFHSJGC	DFHJAVA	1	CICS JVM garbage collection transaction
CJPI	DFHSJPI	DFHJAVA	1	Starts JVMs following a PERFORM JVMPOOL command
CJSR	DFHCJSR	DFHSTAND	1	CICS JVM server resolution transaction
CJTR	DFHOTR	DFHOTS	1	CORBA Object Transaction Services (OTS) resynchronization transaction
CKAM	DFHMQMON	DFHMQ	2	CICS-MQ Adapter alert monitor
CKBM	DFHMQBAS	DFHMQ	2	CICS-MQ Adapter base panel transaction
CKCN	DFHMQQCN	DFHMQ	2	CICS-MQ Adapter start connection transaction

Table 10. CICS transactions supplied by IBM (continued)

Transaction	Program	CSD group	Security category	Description
CKBP	CSQCBP00	DFHMQ	2	CICS-MQ Bridge DPL bridge task
CKDL	DFHMQDSL	DFHMQ	2	CICS-MQ Adapter, display status transaction
CKDP	DFHMQDIS	DFHMQ	2	CICS-MQ Adapter, display transaction
CKQC	DFHMQCTL	DFHMQ	2	CICS-MQ Adapter control transaction
CKRS	DFHMQRS	DFHMQ	2	CICS-MQ Adapter modify transaction
CKRT	DFHMQRET	DFHMQ	2	CICS-MQ Adapter screen return transaction
CKSD	DFHMQDSC	DFHMQ	2	CICS-MQ Adapter stop connection transaction
CKSQ	DFHMQSSQ	DFHMQ	2	CICS-MQ Adapter start/stop CKTI transaction
CKTI	DFHMQTSK	DFHMQ	2	CICS-MQ Adapter - task initiator transaction
CLDM*	DFHLDMAP	DFHOPER	2	CICS load module map
CLER				Language Environment runtime options
CLQ2	DFHLUP	DFHISCT	3	Outbound resynchronization for APPC and MRO
CLR1	DFHZLS1	DFHISCT	3	Inbound CNOS for APPC and MRO
CLR2	DFHLUP	DFHISCT	3	Inbound resynchronization for MRO
CLS1	DFHZLS1	DFHISC	3	Provides ISC LU services model
CLS2	DFHLUP	DFHISC	3	Provides ISC LU services model
CLS3	DFHCLS3	DFHISC	3	ISC LU services model
CLS4	DFHCLS4	DFHISC	3	Manages password expiry
CMAC	DFHCMAC		2	Messages utility
CMPX	DFHMXP	DFHISC	3	Ships ISC local queuing
CMSG*	DFHMSP	DFHMSWIT	2	Message switching
CMTS	DFHZATS	DFHSPI	1	Remote mass delete transaction
COVR	DFHZCOVR	-	1	Provides open z/OS Communications Server retry transaction
CPCT	DFHZPCT	DFHPSSGN		Catalog signed on terminals for persistent session signon retention
CPIA*	DFHPITE	DFHPIPE	2	Invokes CPIS from the terminal

Table 10. CICS transactions supplied by IBM (continued)

Transaction	Program	CSD group	Security category	Description
CPIH	DFHPIDSH	DFHPIPE	2	CICS pipeline HTTP inbound router
CPIL	DFHPILSQ	DFHPIPE	2	SOAP WebSphere MQ inbound listener
CPIQ	DFHPIDSQ	DFHPIPE	2	SOAP WebSphere MQ inbound router
CPIR	DFHPIITL	DFHSTAND	1	Pipeline resolution transaction
CPIS	DFHPIR	DFHPIPE	1	WS-AT transaction that is attached when resynchronization is required
CPLT	DFHSIPLT	-	1	Initializes PLT processing
CPMI	DFHMIRS	DFHISC	2	CICS LU 6.2 synchronization level 1 mirror
CPSS	DFHZSGN	DFHPSSGN	3	Persistent sessions signon
CQPI	DFHCLS5	DFHISC	3	Connection quiesce. Architected transaction (inbound).
CQPO	DFHCLS5	DFHISC	3	Connection quiesce. Architected transaction (outbound).
CQRY	DFHQRY	DFHSTAND	3	Provides ATI query support
CRDR	DFHRD1			Asynchronous Transaction Processing in Put Routine 1
CREA*	DFHADDRM	DFHADST	2	Create REQUESTMODELs for enterprise beans
CREC*	DFHADDRM	DFHADST	2	Create REQUESTMODELs for enterprise beans (read only)
CRLR	DFHRLR	DFHRL	1	Bundle resource resolution transaction
CRMD	DFHZATMD	DFHSPI	1	Provides remote mass delete transaction
CRMF	DFHZATMF	DFHSPI	1	Provides remote mass flag transaction
CRPA	DFHRPAS	DFHRPC	2	ONC/RPC Alias transaction
CRPC	DFHRPC00	DFHRPC	2	ONC/RPC Update transaction
CRPM	DFHRPHS	DFHRPC	2	ONC/RPC Server controller
CRSQ	DFHCRQ	DFHISC	1	Remote schedule purging (ISC)
CRSR	DFHCRS	DFHISC	3	Provides ISC remote scheduler
CRSY	DFHRMSY	DFHRMI	1	Resource manager resynchronization
CRTE*	DFHRTE	DFHISC	2	Transaction routing

Table 10. CICS transactions supplied by IBM (continued)

Transaction	Program	CSD group	Security category	Description
CRTP	DFHZRTP	DFHPSSGN	1	Persistent sessions restart timer transaction
CRTX	-	DFHISC	2	Dynamic transaction routing transaction definition
CSAC	DFHACP	DFHSTAND	3	Provides program abnormal condition
CSCY	DFHCPY	DFHVTAMP	3	Provides 3270 screen print
CSFE*	DFHFEP	DFHFE	2	Terminal test, trace, storage
CSFR	DFHFCRD	-	1	RLS cleanup
CSFU	DFHFCU	DFHOPCLS	1	File open utility
CSGM	DFHGMM	DFHVTAM	2	“Good-morning” signon
CSGX	DFHDLG	DFHDLI		
CSHA	-	-	1	Scheduler services (Autoinstalled by CICS)
CSHQ	DFHSHSY	-	1	Scheduler services domain long running task
CSHR	DFHMIRS	DFHISC	2	Scheduler services remote routing
CSIR 160	DFHCRR			Interregion Session Recovery Program
CSJC	DFHJCBSP	DFHCOMP3		Journal control bootstrap
CSKP	DFHAKP	DFHCOMP5	1	Writes system log activity keypoint
CSLG	DFHZRLG	DFHRSPLG		
CSMI	DFHMIRS	DFHISC	2	Mirror transaction
CSMT	DFHMTPA			Master Terminal Module A
CSM1	DFHMIRS	DFHISC	2	SYSMSG model
CSM2	DFHMIRS	DFHISC	2	Scheduler model
CSM3	DFHMIRS	DFHISC	2	Queue model
CSM5	DFHMIRS	DFHISC	2	DL/I model
CSNC	DFHCRNP	DFHISC	1	Interregion control program (MRO)
CSNE	DFHZNAC	DFHVTAM	1	Provides z/OS Communications Server node error recovery
CSOL	DFHSOL	-	1	TCP/IP listener (Autoinstalled by CICS)
CSOT	DFHMTPA			Master Terminal Module A
CSPG*	DFHTPR	DFHBMS	3	Provides BMS terminal paging
CSPK	DFHPRK	DFHVTAMP	3	Provides 3270 screen print support
CSPP	DFHP3270	DFHHARDC	3	Provides 3270 print support

Table 10. CICS transactions supplied by IBM (continued)

Transaction	Program	CSD group	Security category	Description
CSPQ	DFHTPQ	DFHBMS	1	Terminal page cleanup (BMS)
CSPS	DFHTPS	DFHBMS	3	Schedules BMS terminal paging
CSQC	DFHLGQC	DFHLGQC	1	CICS quiesce after system log failure
CSRK	DFHRKB	DFHVTAMP	3	Provides 3270 screen print - release keyboard
CSRS	DFHZRSP	DFHRSEND	3	Synchronizes 3614 message
CSSC	DFHCSSC			Sign off inactive terminal
CSSF	DFHRTC	DFHISC	3	Cancels CRTE transaction routing session
CSSN	DFHSNP			Sign on program
CSST	DFHMTPA			Master terminal module A
CSSX	DFHDLS	DFHDLI		
CSSY	DFHAPATT	-	1	Provides entry point attach (See note)
CSTA	DFHTAJP			Time of day adjustment program
CSTE	DFHTACP	DFHSTAND	1	Processes terminal abnormal conditions
CSTP	DFHZCSTP	-	1	Provides terminal control transaction (See note)
CSTT	DFHSTKC	DFHCOMP1		Supervisory Statistics Program
CSXM			Exempt (See note)	The transaction used by CICS services to get and free a transaction environment
CSZI	DFHSZRMP	DFHFEPI	1	Front End Programming Interface (FEPI), only active if FEPI installed.
CTIN	DFHZCT1	DFHCLNT	2	CICS Client
CTSD	DFHTSDQ	-	1	Temporary storage delete recoverable queue
CVMI	DFHMIRS	DFHISC	2	CICS LU6.2 synchronization level 1 mirror
CVST	DFHVAP			Subtask Monitor Program
CWBA	DFHWBA	DFHWEB	2	CICS Web support alias transaction
CWBC	DFHWBC00	DFHWEB		CICS Web support connection manager
CWBG	DFHWBGB	DFHWEB	1	CICS Web support cleanup transaction
CWBM	DFHWBM	DFHWEB		CICS Web support server controller
CWTO*	DFHCWTO	DFHCONS	2	Write to console operator

Table 10. CICS transactions supplied by IBM (continued)

Transaction	Program	CSD group	Security category	Description
CWWU	DFHWBA	DFHCURDI	2	Web support alias transaction for the CICS management client interface
CWXN	DFHWBXN	DFHWEB	1	CICS Web support attach transaction
CWXU	DFHWBXN	DFHWEB	1	CICS Web support USER protocol attach transaction
CW2A	DFHW2A	DFHWEB2	2	Atom feed alias transaction
CXCU	DFHCXCU	DFHSTAND	1	Performs XRF tracing catchup
CXRE	DFHZXRE	DFHSTAND	1	Reconnects terminals following XRF takeover
CXRT	DFHCRT	DFHISC	3	Provides Transaction routing relay
DSNC	DFHD2CM1	DFH$DB2	2	DB2 attachment facility transaction

Note:

1. Transactions CGRP, CSSY, CSTP, and CSXM are transaction names that are used by some CICS tasks.
2. Transactions CDBM and CSXN are not subject to security checking, and are exempt from security categorization. Any security definitions for these transactions are redundant.

Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not 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:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply in 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 publication 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.

Licensees of this program who want 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 United Kingdom Laboratories, MP151, Hursley Park, Winchester, Hampshire, England, SO21 2JN.

© Copyright IBM Corp. 1982, 2011

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Programming License Agreement, or any equivalent agreement between us.

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 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.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other product and service names might be trademarks of IBM or other companies.

Bibliography

CICS books for CICS Transaction Server for z/OS

General

CICS Transaction Server for z/OS Program Directory, GI13-0565
CICS Transaction Server for z/OS What's New, GC34-7192
CICS Transaction Server for z/OS Upgrading from CICS TS Version 3.1, GC34-7188
CICS Transaction Server for z/OS Upgrading from CICS TS Version 3.2, GC34-7189
CICS Transaction Server for z/OS Upgrading from CICS TS Version 4.1, GC34-7190
CICS Transaction Server for z/OS Installation Guide, GC34-7171

Access to CICS

CICS Internet Guide, SC34-7173
CICS Web Services Guide, SC34-7191

Administration

CICS System Definition Guide, SC34-7185
CICS Customization Guide, SC34-7161
CICS Resource Definition Guide, SC34-7181
CICS Operations and Utilities Guide, SC34-7213
CICS RACF Security Guide, SC34-7179
CICS Supplied Transactions, SC34-7184

Programming

CICS Application Programming Guide, SC34-7158
CICS Application Programming Reference, SC34-7159
CICS System Programming Reference, SC34-7186
CICS Front End Programming Interface User's Guide, SC34-7169
CICS C++ OO Class Libraries, SC34-7162
CICS Distributed Transaction Programming Guide, SC34-7167
CICS Business Transaction Services, SC34-7160
Java Applications in CICS, SC34-7174

Diagnosis

CICS Problem Determination Guide, GC34-7178
CICS Performance Guide, SC34-7177
CICS Messages and Codes Vol 1, GC34-7175
CICS Messages and Codes Vol 2, GC34-7176
CICS Diagnosis Reference, GC34-7166
CICS Recovery and Restart Guide, SC34-7180
CICS Data Areas, GC34-7163
CICS Trace Entries, SC34-7187
CICS Supplementary Data Areas, GC34-7183
CICS Debugging Tools Interfaces Reference, GC34-7165

Communication

CICS Intercommunication Guide, SC34-7172
CICS External Interfaces Guide, SC34-7168

© Copyright IBM Corp. 1982, 2011

Databases

CICS DB2 Guide, SC34-7164

CICS IMS Database Control Guide, SC34-7170

CICS Shared Data Tables Guide, SC34-7182

CICSPlex SM books for CICS Transaction Server for z/OS

General

CICSPlex SM Concepts and Planning, SC34-7196
CICSPlex SM Web User Interface Guide, SC34-7214

Administration and Management

CICSPlex SM Administration, SC34-7193
CICSPlex SM Operations Views Reference, SC34-7202
CICSPlex SM Monitor Views Reference, SC34-7200
CICSPlex SM Managing Workloads, SC34-7199
CICSPlex SM Managing Resource Usage, SC34-7198
CICSPlex SM Managing Business Applications, SC34-7197

Programming

CICSPlex SM Application Programming Guide, SC34-7194
CICSPlex SM Application Programming Reference, SC34-7195

Diagnosis

CICSPlex SM Resource Tables Reference Vol 1, SC34-7204
CICSPlex SM Resource Tables Reference Vol 2, SC34-7205
CICSPlex SM Messages and Codes, GC34-7201
CICSPlex SM Problem Determination, GC34-7203

Other CICS publications

The following publications contain further information about CICS, but are not provided as part of CICS Transaction Server for z/OS, Version 4 Release 2.

Designing and Programming CICS Applications, SR23-9692

CICS Application Migration Aid Guide, SC33-0768

CICS Family: API Structure, SC33-1007

CICS Family: Client/Server Programming, SC33-1435

CICS Family: Interproduct Communication, SC34-6853

CICS Family: Communicating from CICS on System/390, SC34-6854

CICS Transaction Gateway for z/OS Administration, SC34-5528

CICS Family: General Information, GC33-0155

CICS 4.1 Sample Applications Guide, SC33-1173

CICS/ESA 3.3 XRF Guide , SC33-0661

Other IBM publications

The following publications contain information about related IBM products.

Resource Access Control Facility (RACF): Security Administrator's Guide, SC28-1340

IMS Utilities Reference, SC26-4284

IMS Messages and Code, SC26-4290

IMS System Administration Guide, SC26-4278

IMS Operations Guide, SC26-4287

An Introduction to the IBM 3270 Information Display System, GA27-2739

Systems Network Architecture—LU6.2 Reference: Peer Protocols, SC31-6808

MVS/DFP 3.2 System Programming Reference, SC26-4567

IBM Developer Kit and Runtime Environment, Java 2 Technology Edition, Version 1.4.2 Diagnostics Guide, SC34-6358

Accessibility

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

You can perform most tasks required to set up, run, and maintain your CICS system in one of these ways:

- using a 3270 emulator logged on to CICS
- using a 3270 emulator logged on to TSO
- using a 3270 emulator as an MVS system console

IBM Personal Communications provides 3270 emulation with accessibility features for people with disabilities. You can use this product to provide the accessibility features you need in your CICS system.

© Copyright IBM Corp. 1982, 2011

Index

Special characters

-TERMID*LDC/OPID
  CMSG 586
? and CEMT 113
(value)
  CEMT INQUIRE JMODEL 243
  CEMT INQUIRE JOURNALNAME 246
  CEMT SET PROCESSTYPE 495
* and CEMT 115
*ALL
  CSPG 619
*LDC
  CMSG 586
+ in CEMT syntax 115
+ in scrolling under CEMT 117
+HHMM
  CMSG 590
+M
  CMSG 590
+MM
  CMSG 590
+TERMID*LDC/OPID
  CMSG 586

Numerics

3270 interface
  application debugging profile manager 37

A

abbreviation of keywords
  CEMT transaction 115
abbreviations for keywords 3
ABEND
  CEMT INQUIRE DB2CONN 163
  CEMT INQUIRE DB2ENTRY 176
  CEMT SET DB2CONN 440
  CEMT SET DB2ENTRY 451
access method control block (ACB) 392, 525
ACCESSMETHOD
  CEMT INQUIRE CONNECTION 151
  CEMT INQUIRE DSNAME 197
  CEMT INQUIRE FILE 223
ACCOUNTREC
  CEMT INQUIRE DB2CONN 163
  CEMT INQUIRE DB2ENTRY 174
  CEMT SET DB2CONN 440
  CEMT SET DB2ENTRY 449
ACQ
  CEOT 534
ACQSTATUS
  CEMT INQUIRE MODENAME 262
ACQUIRED
  CEMT INQUIRE CONNECTION 152
  CEMT INQUIRE IPCONN 236
  CEMT INQUIRE MODENAME 262
ACQUIRED *(continued)*
  CEMT INQUIRE NETNAME 276
  CEMT INQUIRE TERMINAL 348
  CEMT SET CONNECTION 433
  CEMT SET IPCONN 474
  CEMT SET MODENAME 484
  CEMT SET NETNAME 491
  CEMT SET TERMINAL 516
acquiring z/OS Communications Server terminals 491, 516
ACTION
  CEMT INQUIRE DSNAME 197
  CEMT INQUIRE SYDUMPCODE 309
  CEMT INQUIRE TRDUMPCODE 357
  CEMT SET TSQUEUE 522
activating debugging profiles
  with the 3270 interface 41
  with the web interface 22
ACTIVE
  CEMT INQUIRE ENQ 207
  CEMT INQUIRE MODENAME 262
  CEMT INQUIRE SYSTEM 316, 317
  CEMT INQUIRE TCLASS 324
  CEMT INQUIRE UOW 372
  CEMT INQUIRE UOWENQ 380
ACTIVITY
  CEMT INQUIRE TASK 318
  CEMT SET PROCESSTYPE 494
ACTIVITYID
  CEMT INQUIRE TASK 319
ACTJVMTCBS
  CEMT INQUIRE DISPATCHER 183
ACTOPENTCBS
  CEMT INQUIRE DISPATCHER 183
ACTSOCKETS
  CEMT INQUIRE TCPIP 326
ACTSSLTCBS
  CEMT INQUIRE DISPATCHER 183
ACTTHRDTCBS
  CEMT INQUIRE DISPATCHER 183
ACTXPTCBS
  CEMT INQUIRE DISPATCHER 183
ADAPTERTYPE option
  CEMT INQUIRE EPADAPTER command 213
ADD
  CEMT INQUIRE SYDUMPCODE 309
  CEMT INQUIRE TRDUMPCODE 358
  CEMT SET SYDUMPCODE 502
  CEMT SET TRDUMPCODE 521
ADDABLE
  CEMT INQUIRE FILE 224
  CEMT SET FILE 467
ADDREXECENQ
  CEMT INQUIRE ENQ 208
  CEMT INQUIRE UOWENQ 380
ADDSTATUS
  CEMT INQUIRE FILE 224
AFFINITY
  CEMT INQUIRE CONNECTION 151
AGE
  CEMT INQUIRE JVM 249
  CEMT INQUIRE UOW 370
AGING
  CEMT INQUIRE DISPATCHER 183
  CEMT INQUIRE SYSTEM 311
  CEMT SET DISPATCHER 455
  CEMT SET SYSTEM 504
AIBRIDGE
  CEMT INQUIRE AUTOINSTALL 133
  CEMT SET AUTOINSTALL 428
AKP
  CEMT INQUIRE SYSTEM 312
  CEMT SET SYSTEM 504
ALL
  CEBT INQUIRE AUXTRACE 78
  CEBT SET AUXTRACE 82
  CEMT INQUIRE ATOMSERVICE 128
  CEMT INQUIRE AUTINSTMODEL 132
  CEMT INQUIRE AUXTRACE 136
  CEMT INQUIRE BEAN 138
  CEMT INQUIRE BRFACILITY 140
  CEMT INQUIRE CONNECTION 149
  CEMT INQUIRE CORBASERVER 158
  CEMT INQUIRE DB2CONN 163
  CEMT INQUIRE DB2ENTRY 173
  CEMT INQUIRE DB2TRAN 178
  CEMT INQUIRE DJAR 186
  CEMT INQUIRE DSNAME 196
  CEMT INQUIRE ENQMODEL 210
  CEMT INQUIRE FILE 221
  CEMT INQUIRE IPCONN 234
  CEMT INQUIRE JMODEL 243
  CEMT INQUIRE JOURNALNAME 247
  CEMT INQUIRE MODENAME 261
  CEMT INQUIRE NETNAME 272
  CEMT INQUIRE PARTNER 277
  CEMT INQUIRE PIPELINE 279
  CEMT INQUIRE PROCESSTYPE 283
  CEMT INQUIRE PROFILE 287
  CEMT INQUIRE PROGRAM 289
  CEMT INQUIRE REQUESTMODEL 297
  CEMT INQUIRE STREAMNAME 306
  CEMT INQUIRE SYDUMPCODE 308
  CEMT INQUIRE TASK 317
  CEMT INQUIRE TCLASS 322
  CEMT INQUIRE TDQUEUE 335
  CEMT INQUIRE TERMINAL 343

© Copyright IBM Corp. 1982, 2011

ALL *(continued)*
CEMT INQUIRE TRANSACTION 350
CEMT INQUIRE TRDUMPCODE 357
CEMT INQUIRE TSMODEL 360
CEMT INQUIRE TSPOOL 363
CEMT INQUIRE TSQUEUE 364
CEMT INQUIRE UOW 368
CEMT INQUIRE UOWLINK 381
CEMT INQUIRE URIMAP 385
CEMT INQUIRE WEBSERVICE 395
CEMT INQUIRE WORKREQUEST 402
CEMT INQUIRE XMLTRANSFORM 406
CEMT PERFORM STATISTICS 422
CEMT SET ATOMSERVICE 426
CEMT SET AUXTRACE 429
CEMT SET BRFACILITY 430
CEMT SET BUNDLE 431
CEMT SET CONNECTION 434
CEMT SET CORBASERVER 439
CEMT SET DOCTEMPLATE 457
CEMT SET DSNAME 460
CEMT SET FILE 467
CEMT SET IPCONN 474
CEMT SET JOURNALNAME 479
CEMT SET MODENAME 484
CEMT SET NETNAME 491
CEMT SET PIPELINE 493
CEMT SET PROCESSTYPE 495
CEMT SET PROGRAM 496
CEMT SET SYDUMPCODE 502
CEMT SET TASK 508
CEMT SET TCLASS 509
CEMT SET TERMINAL 516
CEMT SET TRANSACTION 519
CEMT SET TRDUMPCODE 521
CEMT SET TSQUEUE 522
CEMT SET UOW 523
CEMT SET UOWLINK 524
CEMT SET WEBSERVICE 529
CEMT SET WORKREQUEST 530
CEMT SET XMLTRANSFORM 531
CMSG 578
general statement 115
ALL ROUTE ENTRIES DELETED
CMSG 590
ALL*LDC
CMSG 586
ALLOCAGE
CEMT INQUIRE JVM 249
ANALYZER
CEMT INQUIRE URIMAP 387
ANALYZERSTAT
CEMT INQUIRE URIMAP 386
ANY
CEMT INQUIRE PROGRAM 289
CEMT INQUIRE TRANSACTION 355
APIST
CEMT INQUIRE PROGRAM 289
APPC
CEMT INQUIRE CONNECTION 155
CEMT INQUIRE UOWLINK 383
APPENDCRLF
CEMT INQUIRE DOCTEMPLATE 190
application debugging profile manager 15
3270 interface 37
activating debugging profiles 41
changing debugging profiles 42
copying debugging profiles 43
create Java debugging profile screen 49
create LE debugging profile screen 45
creating debugging profiles 40
deleting debugging profiles 44
inactivating debugging profiles 42
list debugging profiles screen 37
set LE debugging display device screen 56
setting the display device 55
view Java debugging profile screen 53
view LE debugging profile screen 52
viewing debugging profiles 42
web interface 15
activating debugging profiles 22
changing debugging profiles 24
configuring 15
copying debugging profiles 24
Create Java profile page 29
Create LE profile page 26
creating debugging profiles 21
deleting debugging profiles 25
inactivating debugging profiles 23
List profiles page 17
Set LE display device page 35
setting the display device 34
View Java profile page 33
View LE profile page 32
viewing debugging profiles 23
APPLID
CEMT INQUIRE IPCONN 235
APPLNAME
CEMT INQUIRE MONITOR 263
APPLNAMEST
CEMT INQUIRE MONITOR 263
ARCHIVEFILE
CEMT INQUIRE WEBSERVICE 398
ASSEMBLER
CEMT INQUIRE PROGRAM 289
ASSERTED
CEMT INQUIRE CORBASERVER 159
ATI
CEMT INQUIRE NETNAME 273
CEMT INQUIRE TERMINAL 345
CEMT SET NETNAME 491
CEMT SET TERMINAL 517
CEOT 534
ATI (automatic transaction initiation) 341
ATISTATUS
CEMT INQUIRE NETNAME 273
CEMT INQUIRE TERMINAL 345
ATOM
CEMT INQUIRE URIMAP 391
ATOMSERVICE
CEMT DISCARD 122
CEMT INQUIRE transaction 127
CEMT INQUIRE URIMAP 387
CEMT PERFORM STATISTICS 423
CEMT SET ATOMSERVICE 426
ATOMTYPE
CEMT INQUIRE ATOMSERVICE 129
ATTACHSEC 329
AUDITLEVEL
CEMT INQUIRE PROCESSTYPE 284
AUDITLOG
CEMT INQUIRE PROCESSTYPE 285
AUT
CEOT 534
AUTHENTICATE 329
CEMT INQUIRE URIMAP 387
AUTHID
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE DB2ENTRY 174
CEMT SET DB2CONN 440
CEMT SET DB2ENTRY 450
AUTHORITY option
CEMT INQUIRE EPADAPTER command 213
AUTHTYPE
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE DB2ENTRY 174
CEMT SET DB2CONN 440
CEMT SET DB2ENTRY 450
AUTHUSERID option
INQUIRE EPADAPTER command 214
AUTINSTMODEL
CEMT DISCARD 122
CEMT transaction 131
AUTOACTIVE
CEMT INQUIRE SYSTEM 315
CEMT SET SYSTEM 504
AUTOINACTIVE
CEMT INQUIRE SYSTEM 315
CEMT SET SYSTEM 504
AUTOINSTALL
CEMT INQUIRE AUTOINSTALL 133
CEMT PERFORM STATISTICS 423
CEMT SET transaction 427
CEMT transaction 132
CSFE ZCQTRACE 616
AUTOMATIC
CEBT SET TAKEOVER 85
automatic transaction initiation (ATI) 341
AUTOPAGEABLE
CEMT INQUIRE NETNAME 275
CEMT INQUIRE TERMINAL 347
CEMT SET NETNAME 491
CEMT SET TERMINAL 517
CEOT 535
autopaging terminal 627
AUTOPUB
CEMT SET CORBASERVER 439

AUTOPUBLISH
CEMT INQUIRE CORBASERVER 159
AUTOSTARTST
CEMT INQUIRE CLASSCACHE 146
AUTOSWITCH
CEBT INQUIRE DUMP 79
CEBT SET DUMP 83
CEMT INQUIRE DUMPDS 204
CEMT SET DUMPDS 463
AUXILIARY
CEBT INQUIRE AUXTRACE 78
CEMT INQUIRE AUXTRACE 136
CEMT INQUIRE TSMODEL 362
CEMT INQUIRE TSQUEUE 366
auxiliary trace 553
CEBT 78
CEBT SET 81
CEMT 135
CEMT SET 428
AUXTRACE
CEBT 78
CEBT SET transaction 81
CEMT SET transaction 428
CEMT transaction 135
AVAILABLE
CEMT INQUIRE CONNECTION 153
CEMT INQUIRE DSNAME 199
CEMT INQUIRE MODENAME 262
CEMT SET DSNAME 460
CEMT SET MODENAME 484

B

BACKLOG 330
CEMT SET TCPIPSERVICE 511
BACKOUT
CEMT INQUIRE CONNECTION 157
CEMT INQUIRE DSNAME 202
CEMT INQUIRE TASK 320
CEMT INQUIRE TRANSACTION 353
CEMT INQUIRE UOW 371
CEMT SET CONNECTION 434
CEMT SET DSNAME 460
CEMT SET IPCONN 474
CEMT SET UOW 523
BACKUPNONBWO
CEMT INQUIRE UOWDSNFAIL 375
BACKUPTYPE
CEMT INQUIRE DSNAME 199
BASE
CEMT INQUIRE DSNAME 201
BASEDSNAME
CEMT INQUIRE DSNAME 200
BASESCOPE
See CEMT INQUIRE BUNDLE
basic mapping support (BMS)
logical message 622
page retrieval 619
paging commands 619
BASICAUTH
CEMT INQUIRE URIMAP 387
BDAM
CEMT INQUIRE DSNAME 197
CEMT INQUIRE FILE 223
BEAN
CEMT PERFORM STATISTICS 423
BEAN command
CEMT transaction 137
BEAN terminal definitions
CEMT INQUIRE BEAN 137
BEANNAME
CEMT INQUIRE REQUESTMODEL 299
BELOW
CEMT INQUIRE PROGRAM 289
CEMT INQUIRE TRANSACTION 355
BINDFILE
CEMT INQUIRE ATOMSERVICE 129
BINDING
CEMT INQUIRE WEBSERVICE 398
blank fields in a display 117
BMS (basic mapping support) 619
BMS paging
copy command 626
display routed pages 627
logical message 625
status of hardcopy terminal 627
BREXIT
CEMT INQUIRE TRANSACTION 352
BRFACILITY
CEMT INQUIRE TASK 319
CEMT SET transaction 430
BRIDGE
CEMT INQUIRE TASK 319
bridge facilities
CEMT INQUIRE requests 138
Bridge facilities
CEMT SET requests 430
BROWSE
CEMT INQUIRE FILE 224
CEMT SET FILE 467
BROWSESTATUS
CEMT INQUIRE FILE 224
BSAM terminals 577
BTS browser transaction 59
BUNDLE
CEMT DISCARD 123
CEMT INQUIRE transaction 142
CEMT PERFORM STATISTICS 423
CEMT SET transaction 431
BUNDLEDIR
CEMT INQUIRE BUNDLE 143

C

C
CEMT INQUIRE PROGRAM 289
CACHE
CEMT INQUIRE UOWDSNFAIL 374
CACHEFREE
CEMT INQUIRE CLASSCACHE 146
CACHESIZE
CEMT INQUIRE CLASSCACHE 146
CEMT PERFORM CLASSCACHE command 410
CADP transaction 37, 633
CANCEL
CEMT INQUIRE CONNECTION 155
CANCEL *(continued)*
CEMT SET CONNECTION 434
CEMT SET IPCONN 474
CEMT SET TERMINAL 517
CMSG 578
CANCELED - TEMP STRG ERR N
CMSG 590
canceling a message 575
canceling a transaction 2, 575, 576, 577, 578, 629
CAPTURESPEC
CEMT PERFORM STATISTICS 423
CATA transaction 633
CATD transaction 633
CATEGORY
CEMT INQUIRE ATOMSERVICE 129
CATR transaction 633
CATS transaction 633
CAUSE
CEMT INQUIRE UOWDSNFAIL 374
CAUT transaction 633
CAVM (CICS availability manager) 86
CBAM transaction 59, 633
CBRC transaction 633
CCIN transaction 633
CCMF transaction 633
CCRL transaction 61, 633
CCSID
CEMT INQUIRE WEBSERVICE 398
CEMT INQUIRE XMLTRANSFORM 407
CDATAKEY
CEMT INQUIRE TRANSACTION 355
CDBC transaction 63, 633
CDBD transaction 633
CDBF transaction 633
CDBI transaction 67, 633
CDBM transaction 69, 633
CDBN transaction 633
CDBO transaction 633
CDBQ transaction 633
CDBT transaction 634
CDFS transaction 634
CDSASIZE
CEMT INQUIRE DSAS 193
CDST transaction 634
CDTS transaction 634
CEBR transaction 75, 634
CEBT transaction 634
AUXTRACE 78, 81
CICS availability manager (CAVM) 86
DUMP 79, 82
GTFTRACE 79, 83
INQ|SET 77
INTTRACE 80, 84
PERFORM 86
request formats 77
SHUTDOWN 86
SNAP 86
SURVEILLANCE 81, 85
TAKEOVER 81, 85, 87
XRF complex 77
CECI transaction 89, 634
CECS transaction 89, 634

CEDA transaction 91, 634
  security checking 96
CEDB transaction 91, 634
CEDC transaction 91, 634
CEDF
  CEMT INQUIRE PROGRAM 289
  CEMT SET PROGRAM 496
CEDF transaction 99, 634
CEDFSTATUS
  CEMT INQUIRE PROGRAM 289
CEDX transaction 634
CEGN transaction 634
CEHP transaction 634
CEHS transaction 634
CEJR transaction 634
CEKL transaction 634
  request formats 105
  TASK 105, 108
CEMN monitoring facility transaction 111
CEMN transaction 111, 634
CEMS transaction 634
CEMT INQUIRE CONNECTION 157
CEMT PERFORM CLASSCACHE command 409
CEMT PERFORM JVMPOOL command 416
CEMT PERFORM PIPELINE command 417
CEMT SET EVENTPROCESS 466
CEMT SET transaction
  BUNDLE 431
  EPADAPTER 464
  EVENTBINDING 465
  EVENTPROCESS 465
CEMT transaction 634
  ? character 113
  * symbol 115
  + in CEMT syntax 115
  + in scrolling 117
  abbreviation of keywords 115
  ATOMSERVICE 127, 426
  AUTINSTMODEL 131
  AUTOINSTALL 132, 427
  AUXTRACE 135, 428
  BEAN 137
  blank fields in a display 117
  BRFACILITY 138, 430
  BUNDLE 142
  CEMT on the command line 113
  CFDTPOOL 144
  CLASSCACHE 146, 432
  CONNECTION 148, 432
  CORBASERVER 158, 439
  DB2CONN 163, 440
  DB2ENTRY 448
  DB2TRAN 178, 452
  DBENTRY 173
  DELETSHIPPED 181, 453
  DISCARD 121, 122
  DISPATCHER 182, 454
  DJAR 185
  DOCTEMPLATE 189, 456
  DSAS 193, 457
  DSNAME 195, 458
  DUMP 414
  DUMPDS 203, 462
CEMT transaction *(continued)*
  ENDAFFINITY 415
  ENQ 205
  ENQMODEL 209
  ENQQUEUE 463
  EPADAPTER 212
  EVENTBINDING 216
  EVENTPROCESS 218
  EXCI 219
  Family of resources 115
  FILE 221, 466
  generic names 115
  generic resource identifiers 115
  GTFTRACE 230, 471
  HOST 231, 472
  INQUIRE 114
  INQUIRE IPCONN 233
  INQUIRE LIBRARY 254
  INTTRACE 232, 473
  invoking from an application program 120
  IPCONN 473
  IRBATCH 241
  IRC 241, 477
  JMODEL 243
  JOURNALNAME 246, 478
  JVM 248
  JVMPOOL 250, 479
  JVMSERVER 251, 480
  LINE 259, 482
  minus sign 113
  MODENAME 261, 483
  MONITOR 263, 484
  MQCONN 266, 488
  MQINI 269
  NETNAME 271, 490
  overtyping a display 117
  PARTNER 277
  PERFORM 408
  PIPELINE 492
  PROCESSTYPE 494
  PROFILE 286
  PROGRAM 289, 495
  program function keys 114
  question mark preceding CEMT 113
  QUEUE 498
  request formats 113
  REQUESTMODEL 297
  RESET 418
  RRMS 301
  scrolling a display 117
  SECURITY 418
  security checking 120
  SET 114
  SHUTDOWN 419
  SNAP 414
  STATISTICS 302, 420, 498
  STREAMNAME 305
  SYDUMPCODE 307, 501
  SYSTEM 310, 503
  tab key 117
  TASK 317, 507
  TCLASS 322, 509
  TCPIP 325, 509
  TCPIPSERVICE 327, 511
  TDQUEUE 335, 512
  TEMPSTORAGE 341, 514
CEMT transaction *(continued)*
  TERMINAL 342, 515
  TRANSACTION 349, 519
  TRDUMPCODE 356, 520
  TSMODEL 359
  TSPOOL 362
  TSQNAME 364
  TSQUEUE 364, 522
  UOW 368, 523
  UOWDSNFAIL 373
  UOWENQ 377
  UOWLINK 381, 524
  URIMAP 385, 524
  WEB 394, 528
  WEBSERVICE 528
  without keywords 114
  WORKREQUEST 401, 529
  XMLTRANSFORM 405, 531
  z/OS Communications Server 392, 525
CEOS transaction 634
CEOT transaction 533, 634
CEPD transaction 634
CEPF transaction 634
CEPH transaction 634
CEPM transaction 634
CEPQ transaction 635
CEPT transaction 635
CERTIFICATE 330
  CEMT INQUIRE CORBASERVER 160
  CEMT INQUIRE URIMAP 387
CERTIFICATE option
  CEMT INQUIRE IPCONN command 235
certificate revocation list transaction (CCRL) 61
CESC transaction 635
CESD transaction 635
CESF transaction 537, 635
CESL
  using CESN as an alias 548
CESL transaction 635
  customizing map set 539
CESN 542, 543, 547, 548
  and password phrases 548
CESN transaction 545, 635
  customizing map set 545
CEST transaction 551, 635
CETR transaction 553, 635
CEX2 transaction 635
CEXECKEY
  CEMT INQUIRE PROGRAM 289
  CEMT PERFORM JVMPOOL command 416
CFCL transaction 635
CFDTPOOL
  CEMT INQUIRE CFDTPOOL 145
  CEMT SET FILE 467
  CEMT transaction 144
CFOR transaction 635
CFQR transaction 635
CFQS transaction 635
CFTABLE
  CEMT INQUIRE FILE 229
  CEMT INQUIRE UOWLINK 384
  CEMT SET FILE 467

CFTI transaction 635
CFTL transaction 635
CFTM transaction 635
CFTS transaction 635
CFUP transaction 635
CGROUP
  CEMT INQUIRE DB2CONN 163
  CEMT SET DB2CONN 440
CGRP transaction 635
chaining a message
  CSPG transaction 623
  partitions 7
changing debugging profiles
  with the 3270 interface 42
  with the web interface 24
CHARACTERSET
  CEMT INQUIRE URIMAP 388
CHLP transaction 635
CICS availability manager (CAVM) 86
CICS DB2 transaction (DSNC) 631
CICS releases, CSD compatibility
  across 95
CICSAPI
  CEMT INQUIRE PROGRAM 289
CICSTABLE
  CEMT INQUIRE FILE 229
  CEMT SET FILE 467
CICSTELEVEL
  CEMT INQUIRE SYSTEM 312
CID
  CEMT INQUIRE NETNAME 273
  CEMT INQUIRE TERMINAL 345
CIDDOMAIN
  CEMT INQUIRE PIPELINE 280
CIDP transaction 559, 635
CIEP transaction 635
CIND transaction 561, 635
  running 562
CINS transaction 635
CIOD transaction 635
CIOF transaction 635
CIOR transaction 635
CIPHERS
  CEMT INQUIRE URIMAP 388
CIPHERS option
  CEMT INQUIRE IPCONN
    command 236
CIRB transaction 636
CIRD transaction 636
CIRP transaction 636
CIRR transaction 636
CIS4 transaction 636
CISB transaction 636
CISC transaction 636
CISD transaction 636
CISE transaction 636
CISM transaction 636
CISQ transaction 636
CISR transaction 636
CISS transaction 636
CIST transaction 636
CISU transaction 636
CISX transaction 636
CITS transaction 636
CJGC transaction 636
CJPI transaction 636
CJSR transaction 636
CJTR transaction 636
CKAM transaction 636
CKBM transaction 636
CKBP transaction 637
CKCN transaction 636
CKDL transaction 637
CKDP transaction 637
CKOPEN
  CEMT INQUIRE TASK 321
CKQC transaction 637
CKRS transaction 637
CKRT transaction 637
CKSD transaction 637
CKSQ transaction 637
CKTI transaction 637
CLASS
  CEMT INQUIRE PROGRAM 289
  CEMT INQUIRE TERMINAL 343
  CEMT INQUIRE
    TRANSACTION 350
  CEMT SET PROGRAM 496
  CEMT SET TERMINAL 517
  CEMT SET TRANSACTION 519
  general statement 115
class tasks, CEMT requests 322, 509
CLASSCACHE
  CEMT SET CLASSCACHE 432
  CEMT transaction 146, 409
CLASSCACHEST
  CEMT INQUIRE JVM 249
CLDM transaction 565, 637
CLEAR key
  when invoking transaction 2
  when used with partitions 7
CLEAR PARTITION key 7
CLEAR/CANCL or CONT
  INPUT/ROUTE/MSG
  CMSG 590
CLER transaction 567, 637
CLIENT
  CEMT INQUIRE URIMAP 391
CLIENTCERT
  CEMT INQUIRE
    CORBASERVER 160
CLIENTIPADDR option
  INQUIRE WORKREQUEST
    command 403
CLNTIP6ADDR option
  INQUIRE WORKREQUEST
    command 403
CLNTIPFAMILY option
  INQUIRE WORKREQUEST
    command 403
CLOSED
  CEBT INQUIRE DUMP 79
  CEBT SET DUMP 83
  CEMT INQUIRE DUMPDS 204
  CEMT INQUIRE FILE 228
  CEMT INQUIRE IRC 242
  CEMT INQUIRE MODENAME 262
  CEMT INQUIRE RRMS 302
  CEMT INQUIRE TDQUEUE 340
  CEMT INQUIRE VTAM 393
  CEMT SET DUMPDS 463
  CEMT SET FILE 467
  CEMT SET IRC 478
  CEMT SET MODENAME 484
CLOSED (continued)
  CEMT SET VTAM 526
CLOSETIMEOUT
  INQUIRE TCPIPSERVICE 331
CLQ2 transaction 637
CLR1 transaction 637
CLR2 transaction 637
CLS1 transaction 637
CLS2 transaction 637
CLS3 transaction 637
CLS4 transaction 637
CMAC transaction 569, 637
CMDPROT
  CEMT INQUIRE SYSTEM 312
CMDPROTECT
  CEMT INQUIRE SYSTEM 312
CMPX transaction 637
CMSG transaction 573, 637
  CANCEL 578
  DATE 578
  ERRTERM 578
  FULLDATE 578
  HEADING 578
  ID 578
  logical unit destination 586
  MSG 578
  OPCLASS 578
  options 578
  program (DFHMSP) 590
  PROTECT 578
  ROUTE 578
  SEND 578
  syntax 578
  TIME 578
CMTS transaction 637
COBOL
  CEMT INQUIRE PROGRAM 289
codes, display 569
COLD
  CEMT INQUIRE UOWLINK 383
COLDACQ
  CEMT INQUIRE NETNAME 276
  CEMT INQUIRE TERMINAL 349
  CEMT SET NETNAME 491
  CEMT SET TERMINAL 517
COLLECTION
  CEMT INQUIRE
    ATOMSERVICE 129
COMAUTHID
  CEMT INQUIRE DB2CONN 163
  CEMT SET DB2CONN 440
COMAUTHTYPE
  CEMT INQUIRE DB2CONN 163
COMMAND
  CEBT SET TAKEOVER 85
command interpreter transaction
  (CECI) 89
command syntax checker (CECS) 89
commands
  CEBT INQ|SET 77
  CEBT PERFORM 86
  CEMT DISCARD 122
  CEMT INQUIRE 127
  CEMT PERFORM 408
  CEMT SET 425
  paging 619

COMMIT
CEMT INQUIRE CONNECTION 157
CEMT INQUIRE DSNAME 202
CEMT INQUIRE TASK 320
CEMT INQUIRE TRANSACTION 353
CEMT INQUIRE UOW 371
CEMT SET CONNECTION 434
CEMT SET DSNAME 460
CEMT SET IPCONN 475
CEMT SET UOW 523
COMMITFAIL
CEMT INQUIRE UOWDSNFAIL 375
communicating with CICS 1
compatibility, CSD, across CICS releases 95
COMPLETE
CEMT INQUIRE CONNECTION 153
component trace options 553
COMPRESS
CEMT INQUIRE MONITOR 263
CEMT SET MONITOR 486
COMPRESSST
CEMT INQUIRE MONITOR 263
COMTHREADLIM
CEMT INQUIRE DB2CONN 163
CEMT SET DB2CONN 440
COMTHREADS
CEMT INQUIRE DB2CONN 163
CONCURRENCY
CEMT INQUIRE PROGRAM 289
CONFIGDATA1 option
CEMT INQUIRE EPADAPTER command 214
CONFIGFILE
CEMT INQUIRE ATOMSERVICE 130
CEMT INQUIRE PIPELINE 280
CONNECT
CEMT INQUIRE DB2CONN 163
CONNECTED
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE MQCONN 266
CEMT INQUIRE TSPOOL 363
CEMT SET DB2CONN 440
CONNECTERROR
CEMT INQUIRE DB2CONN 163
CEMT SET DB2CONN 440
CONNECTING
CEMT INQUIRE MQCONN 266
CONNECTION
CEMT DISCARD 123
CEMT INQUIRE CONNECTION 152
CEMT INQUIRE MODENAME 262
CEMT INQUIRE UOW 372
CEMT INQUIRE UOWDSNFAIL 374
CEMT INQUIRE UOWLINK 384
CEMT PERFORM STATISTICS 423
CEMT SET CONNECTION 432
CEMT SET MODENAME 484
CONNECTST
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE MQCONN 266
CEMT SET DB2CONN 440
CEMT SET MQCONN 488
CONNSTATUS
CEMT INQUIRE CFDTPOOL 145
CONNSTATUS *(continued)*
CEMT INQUIRE CONNECTION 152
CEMT INQUIRE IPCONN 236
CEMT INQUIRE TSPOOL 363
CONSOLE
CEMT INQUIRE TERMINAL 346
console as a CICS terminal 9
console operator, write to 629
console support, multiple 9
CONSOLES
CEMT INQUIRE AUTOINSTALL 133
CEMT SET AUTOINSTALL 428
CONTAINER
CEMT INQUIRE WEBSERVICE 398
CONTENTION
CEMT SET FILE 468
CONTINUE INPUT
CMSG 590
CONTINUE INPUT OR MSG
CMSG 590
CONTINUE MSG
CMSG 590
CONTINUE ROUTE OPERAND
CMSG 590
CONVERSE
CEMT INQUIRE MONITOR 264
CEMT SET MONITOR 486
CONVERSEST
CEMT INQUIRE MONITOR 264
CONVERTER
CEMT INQUIRE URIMAP 388
COORDINATOR
CEMT INQUIRE UOWLINK 384
COPID
CEMT INQUIRE DB2CONN 163
CEMT SET DB2CONN 440
option of DSNCRCT macro 440
COPY command 626
copying debugging profiles
with the 3270 interface 43
with the web interface 24
copying pages
CSPG 626
partitions 7
COPYSTATUS
CEMT INQUIRE PROGRAM 289
CORBASERVER
CEMT DISCARD 123
CEMT INQUIRE BEAN 138
CEMT INQUIRE DJAR 187
CEMT INQUIRE REQUESTMODEL 300
CEMT PERFORM CORBASERVER 412
CEMT PERFORM STATISTICS 423
CEMT SET CORBASERVER 439
CEMT transaction 158, 411
CORBASERVER option
INQUIRE WORKREQUEST command 403
COVR transaction 637
CPCT transaction 637
CPIA transaction 599
CPIH transaction 638
CPIL transaction 638
CPIQ transaction 638
CPIR transaction 638
CPIS transaction 638
CPLT transaction 638
CPMI transaction 638
CPSS transaction 638
CQP
CEMT INQUIRE CONNECTION 153
CQPI transaction 638
CQPO transaction 638
CQRY transaction 638
CRDR transaction 638
CRE
CEOT 534
CREA transaction 601, 638
CREATE
CEMT INQUIRE NETNAME 274
CEMT INQUIRE TERMINAL 346
CEMT SET NETNAME 491
CEMT SET TERMINAL 517
create Java debugging profile screen
application debugging profile manager
3270 interface 49
Create Java profile page
application debugging profile manager
web interface 29
create LE debugging profile screen
application debugging profile manager
3270 interface 45
Create LE profile page
application debugging profile manager
web interface 26
CREATESESS
CEMT INQUIRE NETNAME 274
CEMT INQUIRE TERMINAL 346
creating debugging profiles
with the 3270 interface 40
with the web interface 21
CREC transaction 601, 638
CRITICAL value 331
CRITICALST option
INQUIRE LIBRARY command 257
CRLR transaction 638
CRMD transaction 638
CRMF transaction 638
CRPA transaction 638
CRPC transaction 638
CRPM transaction 638
CRSQ transaction 638
CRSR transaction 638
CRSY transaction 638
CRTE transaction 613, 638, 639
CRTX transaction 639
CSAC transaction 639
CSCY transaction 639
CSD compatibility across CICS releases 95
CSFE transaction 615, 639
CSFR transaction 639
CSFU transaction 639
CSGM transaction 639
CSGX transaction 639
CSHA transaction 639
CSHQ transaction 639

CSHR transaction 639
CSIGN
CEMT INQUIRE DB2CONN 163
CEMT SET DB2CONN 440
CSIR 160 transaction 639
CSJC transaction 639
CSKP transaction 639
CSLG transaction 639
CSM1 transaction 639
CSM2 transaction 639
CSM3 transaction 639
CSM5 transaction 639
CSMI transaction 639
CSMT transaction 639
CSNC transaction 639
CSNE transaction 639
CSOL transaction 639
CSOT transaction 639
CSPG transaction 619, 639
CSPK transaction 639
CSPP transaction 639
CSPQ transaction 640
CSPS transaction 640
CSQC transaction 640
CSRK transaction 640
CSRS transaction 640
CSSC transaction 640
CSSF transaction 640
CSSN transaction 640
CSST transaction 640
CSSX transaction 640
CSSY transaction 640
CSTA transaction 640
CSTE transaction 640
CSTP transaction 640
CSTT transaction 640
CSXM transaction 640
CSZI transaction 640
CTERM
CEMT INQUIRE DB2CONN 163
CEMT SET DB2CONN 440
DSNCRCT macro 440
CTIN transaction 640
CTLGALL
CEMT INQUIRE SYSTEM 314
CEMT SET SYSTEM 505
CTLGMODIFY
CEMT INQUIRE SYSTEM 314
CEMT SET SYSTEM 505
CTLGNONE
CEMT INQUIRE SYSTEM 314
CEMT SET SYSTEM 505
CTSD transaction 640
CTX
CEMT INQUIRE DB2CONN 163
CEMT SET DB2CONN 440
CURAUXDS
CEMT INQUIRE AUXTRACE 136
CURRENT
CEMT INQUIRE SYDUMPCODE 309
CEMT INQUIRE TRDUMPCODE 358
CURRENT(A|B)
CEBT INQUIRE AUXTRACE 78
CURRENTDDS
CEBT INQUIRE DUMP 79
CURRENTDDS *(continued)*
CEMT INQUIRE DUMPDS 204
CURRENTHEAP
CEMT INQUIRE JVMSERVER 252
CURREQS
CEMT INQUIRE AUTOINSTALL 134
CUSERID
CEMT INQUIRE DB2CONN 163
CEMT SET DB2CONN 440
CVDA values
CBE
CEMT INQUIRE EPADAPTER command 215
CBER
CEMT INQUIRE EPADAPTER command 215
CCE
CEMT INQUIRE EPADAPTER command 215
CFE
CEMT INQUIRE EPADAPTER command 215
CONTEXT
CEMT INQUIRE EPADAPTER command 213
CUSTOM
CEMT INQUIRE EPADAPTER command 213, 214
HTTP
CEMT INQUIRE EPADAPTER command 213, 214
REGION
CEMT INQUIRE EPADAPTER command 213
TRANSTART
CEMT INQUIRE EPADAPTER command 213, 214
TSQ
CEMT INQUIRE EPADAPTER command 213
TSQUEUE 213
CEMT INQUIRE EPADAPTER command 214
USER
CEMT INQUIRE EPADAPTER command 215
USERID
CEMT INQUIRE EPADAPTER command 214
WBE
CEMT INQUIRE EPADAPTER command 215
WMQ
CEMT INQUIRE EPADAPTER command 213, 214
CVMI transaction 640
CVST transaction 640
CW2A transaction 641
CWBA transaction 640
CWBC transaction 640
CWBG transaction 640
CWBM transaction 640
CWTO transaction 629, 640
CWWU transaction 641
CWXN transaction 641
CWXU transaction 641
CXCU transaction 641
CXRE transaction 641
CXRT transaction 641

D

D
CEMT INQUIRE TASK 321
DAE
CEMT INQUIRE SYDUMPCODE 309
CEMT SET SYDUMPCODE 502
DAEOPTION
CEMT INQUIRE SYDUMPCODE 309
data sets
CEMT requests 221
CEMT SET requests 466
data-value
CEMT INQUIRE LIBRARY 255
database control (DBCTL)
connection to 63
disconnection from 63
database control inquiry transaction (CDBI) 67
database control interface transaction (CDBM) 69
database control menu transaction (CDBC) 63
DATABUFFERS
CEMT INQUIRE TDQUEUE 338
DATAFORMAT option
CEMT INQUIRE EPADAPTER command 215
DATALOC
CEMT INQUIRE PROGRAM 289
DATASET
CEMT INQUIRE ENQ 208
CEMT INQUIRE UOW 372
CEMT INQUIRE UOWDSNFAIL 374
CEMT INQUIRE UOWENQ 380
DATASETFULL
CEMT INQUIRE UOWDSNFAIL 375
DATE
CMSG 578
DATE ALREADY PASSED
CMSG 590
DATE INVALID AT XXXXXXXX
CMSG 590
DATE NOT XX/XX AT XXXXX
CMSG 590
date resetting 418
DATE TOO FAR IN FUTURE
CMSG 590
DATESTAMP
CEMT INQUIRE DJAR 187
DATFORM, system initialization parameter 6
DB2
CEMT PERFORM STATISTICS 423
DB2 definitions
connection
CEMT INQUIRE DB2CONN 163
CEMT SET DB2CONN command 440
entry
CEMT INQUIRE DB2ENTRY 173

DB2 definitions *(continued)*
entry *(continued)*
CEMT SET DB2ENTRY command 448
transaction
CEMT INQUIRE DB2TRAN 178
CEMT SET DB2TRAN command 452
DB2CONN
CEMT DISCARD 123
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE SYSTEM 312
CEMT SET transaction 440
CEMT transaction 163
DB2ENTRY
CEMT DISCARD 123
CEMT INQUIRE DB2ENTRY 173
CEMT INQUIRE DB2TRAN 179
CEMT SET DB2TRAN 453
CEMT SET transaction 448
CEMT transaction 173
DB2GROUPID
CEMT INQUIRE DB2CONN 163
CEMT SET DB2CONN 440
DB2ID
CEMT INQUIRE DB2CONN 163
CEMT SET DB2CONN 440
DB2PLAN
CEMT INQUIRE TASK 319
DB2RELEASE
CEMT INQUIRE DB2CONN 163
DB2TRAN
CEMT DISCARD 123
CEMT INQUIRE DB2TRAN 178
CEMT SET transaction 452
CEMT transaction 178
DBCTL (database control) 63
DBCTL system identifier (DBCTLID) 65
DBCTLID 65
DDNAME
CEMT INQUIRE DOCTEMPLATE 191
CEMT INQUIRE TDQUEUE 338
DEADLOCK
CEMT INQUIRE UOWDSNFAIL 375
DEBUG
CEMT SET SYSTEM 505
debugging profile manager
3270 interface 37
activating debugging profiles 41
changing debugging profiles 42
copying debugging profiles 43
create Java debugging profile screen 49
create LE debugging profile screen 45
creating debugging profiles 40
deleting debugging profiles 44
inactivating debugging profiles 42
list debugging profiles screen 37
set LE debugging display device screen 56
setting the display device 55
view Java debugging profile screen 53
debugging profile manager *(continued)*
3270 interface *(continued)*
view LE debugging profile screen 52
viewing debugging profiles 42
web interface 15
activating debugging profiles 22
changing debugging profiles 24
configuring 15
copying debugging profiles 24
Create Java profile page 29
Create LE profile page 26
creating debugging profiles 21
deleting debugging profiles 25
inactivating debugging profiles 23
List profiles page 17
Set LE display device page 35
setting the display device 34
View Java profile page 33
View LE profile page 32
viewing debugging profiles 23
debugging profile manager, application 15
debugging profiles
activating
with the 3270 interface 41
with the web interface 22
changing
with the 3270 interface 42
with the web interface 24
copying
with the 3270 interface 43
with the web interface 24
creating
with the 3270 interface 40
with the web interface 21
deleting
with the 3270 interface 44
with the web interface 25
inactivating
with the 3270 interface 42
with the web interface 23
viewing
with the 3270 interface 42
with the web interface 23
DEBUGTOOL
CEMT INQUIRE SYSTEM 312
DEL
CEMT INQUIRE DELETSHIPPED 181
DELETE
CEMT INQUIRE FILE 225
CEMT SET FILE 468
CEMT SET TSQUEUE 522
CEMT SET UOWLINK 524
DELETESTATUS
CEMT INQUIRE FILE 225
deleting debugging profiles
with the 3270 interface 44
with the web interface 25
DELETSHIPPED
CEMT SET transaction 453
CEMT transaction 413
DELETSHIPPED command
CEMT transaction 181
DELEXITERROR
CEMT INQUIRE UOWDSNFAIL 375
DEREGERROR
INQUIRE TCPIPSERVICE 331
DEREGISTERED
CEMT INQUIRE VTAM 392
CEMT SET VTAM 526
INQUIRE TCPIPSERVICE 331
DEST
CEMT INQUIRE TASK 319
destinations, multiple 573
DFHCICSP 3
DFLTUSER
CEMT INQUIRE SYSTEM 312
DISABLED
CEMT INQUIRE CORBASERVER 161
CEMT INQUIRE DB2ENTRY 176
CEMT INQUIRE ENQMODEL 212
CEMT INQUIRE FILE 226
CEMT INQUIRE HOST 232
CEMT INQUIRE JOURNALNAME 248
CEMT INQUIRE PROGRAM 289
CEMT INQUIRE TDQUEUE 339
CEMT INQUIRE TRANSACTION 355
CEMT INQUIRE URIMAP 388
CEMT PERFORM CLASSCACHE command 410
CEMT SET BUNDLE 431
CEMT SET CLASSCACHE command 432
CEMT SET CORBASERVER 439
CEMT SET DB2ENTRY 451
CEMT SET ENQMODEL 464
CEMT SET EPADAPTER 465
CEMT SET EVENTBINDING 465
CEMT SET FILE 468
CEMT SET HOST 472
CEMT SET JOURNALNAME 479
CEMT SET JVMPOOL 480
CEMT SET PROCESSTYPE 495
CEMT SET PROGRAM 496
CEMT SET TRANSACTION 519
CEMT SET URIMAP 525
DISABLEDACT
CEMT INQUIRE DB2ENTRY 176
CEMT SET DB2ENTRY 451
DISABLING
CEMT INQUIRE CORBASERVER 161
CEMT INQUIRE DB2ENTRY 176
DISCARDING
CEMT INQUIRE CORBASERVER 161
DISCONNING
CEMT INQUIRE MQCONN 266
DISPATCHABLE
CEKL INQUIRE TASK 106
CEMT INQUIRE TASK 321
DISPATCHER
CEMT INQUIRE transaction 182
CEMT PERFORM STATISTICS 423
CEMT SET transaction 454

display device
setting
with the 3270 interface 55
with the web interface 34
display logical message (BMS) 622
display routed pages 627
display transaction 569
DISPOSITION
CEMT INQUIRE FILE 225
CEMT INQUIRE TDQUEUE 338
DJAR
CEMT DISCARD 124
CEMT INQUIRE BEAN 138
CEMT PERFORM DJAR 414
CEMT transaction 185, 413
CREA 602
DJARDIR
CEMT INQUIRE CORBASERVER 161
DNSGROUP
INQUIRE TCPIPSERVICE 331
DNSSTATUS
CEMT SET TCPIPSERVICE 512
DNSSTATUS value
INQUIRE TCPIPSERVICE 331
DOCTEMPLATE
CEMT DISCARD 124
CEMT INQUIRE DOCTEMPLATE 191
CEMT PERFORM STATISTICS 423
CEMT SET transaction 456
CEMT transaction 189
DPLLIMIT
CEMT INQUIRE MONITOR 264
CEMT SET MONITOR 486
DPLSUBSET
CEMT INQUIRE PROGRAM 289
CEMT SET PROGRAM 496
DRAIN
CEMT SET EVENTPROCESS 466
DROLLBACK
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE DB2ENTRY 176
DS
CEMT INQUIRE TASK 321
DSALIMIT
CEMT INQUIRE DSAS 193
CEMT INQUIRE SYSTEM 313
CEMT SET DSAS 458
CEMT SET SYSTEM 505
DSAS
CEMT INQUIRE transaction 193
CEMT SET transaction 457
DSNAME
CEMT INQUIRE DOCTEMPLATE 191
CEMT INQUIRE DSNAME 200
CEMT INQUIRE FILE 225
CEMT INQUIRE TDQUEUE 338
CEMT INQUIRE UOWDSNFAIL 375
CEMT SET FILE 468
CEMT SET transaction 458
CEMT transaction 195
DSNAME option
INQUIRE LIBRARY command 258
DSNC transaction 631, 641
DSNC transactions
DISCONNECT 631
DISPLAY 631
MODIFY 631
STOP 631
STRT 631
DSRTPROGRAM
CEMT SET SYSTEM 505
DTRPROGRAM
CEMT INQUIRE SYSTEM 313
CEMT SET SYSTEM 505
DUMMY
CEMT INQUIRE JMODEL 246
CEMT INQUIRE JOURNALNAME 248
DUMP
CEBT 79
CEBT INQUIRE DUMP 79
CEBT PERFORM SHUTDOWN 86
CEBT PERFORM TAKEOVER 87
CEBT SET transaction 82
CEMT PERFORM DUMP 414
CEMT PERFORM SHUTDOWN 419
CEMT transaction 414
DUMPDS
CEMT INQUIRE DUMPDS 204
CEMT SET transaction 462
CEMT transaction 203
DUMPING
CEMT INQUIRE SYSTEM 313
DUMPSCOPE
CEMT INQUIRE SYDUMPCODE 310
CEMT INQUIRE TRDUMPCODE 358
DURATION
CEMT INQUIRE ENQ 206
CEMT INQUIRE UOWENQ 379
DYNAMIC
CEMT INQUIRE DSNAME 199
CEMT INQUIRE PROGRAM 289
DYNAMSTATUS
CEMT INQUIRE PROGRAM 289

E

ECDSASIZE
CEMT INQUIRE DSAS 194
EDF (execution diagnostic facility) 99
EDSALIMIT
CEMT INQUIRE DSAS 194
CEMT INQUIRE SYSTEM 313
CEMT SET DSAS 458
CEMT SET SYSTEM 505
EMITMODE
CEMT INQUIRE EVENTBINDING 215
EMPTYREQ
CEMT INQUIRE FILE 225
CEMT SET FILE 468
EMPTYSTATUS
CEMT INQUIRE FILE 225
ENABLED
CEMT INQUIRE CORBASERVER 161
CEMT INQUIRE DB2ENTRY 176
CEMT INQUIRE ENQMODEL 212
ENABLED *(continued)*
CEMT INQUIRE FILE 225
CEMT INQUIRE HOST 232
CEMT INQUIRE JOURNALNAME 248
CEMT INQUIRE PROGRAM 289
CEMT INQUIRE TDQUEUE 339
CEMT INQUIRE TRANSACTION 355
CEMT INQUIRE URIMAP 388
CEMT PERFORM CLASSCACHE command 411
CEMT SET BUNDLE 431
CEMT SET CLASSCACHE command 432
CEMT SET CORBASERVER 440
CEMT SET DB2ENTRY 451
CEMT SET ENQMODEL 464
CEMT SET EPADAPTER 465
CEMT SET EVENTBINDING 465
CEMT SET FILE 468
CEMT SET HOST 472
CEMT SET JOURNALNAME 479
CEMT SET JVMPOOL 480
CEMT SET PROCESSTYPE 495
CEMT SET PROGRAM 496
CEMT SET TRANSACTION 520
CEMT SET URIMAP 525
ENABLEDCOUNT
CEMT INQUIRE BUNDLE 144
ENABLESTATUS
CEMT INQUIRE ATOMSERVICE 130
CEMT INQUIRE AUTOINSTALL 134
CEMT INQUIRE BUNDLE 144
CEMT INQUIRE CORBASERVER 161
CEMT INQUIRE DB2ENTRY 176
CEMT INQUIRE EPADAPTER 215
CEMT INQUIRE EVENTBINDING 218
CEMT INQUIRE FILE 225
CEMT INQUIRE HOST 232
CEMT INQUIRE PIPELINE 280
CEMT INQUIRE PROCESSTYPE 285
CEMT INQUIRE TDQUEUE 339
CEMT INQUIRE URIMAP 388
CEMT INQUIRE XMLTRANSFORM 408
CEMT SET ATOMSERVICE 427
CEMT SET DB2ENTRY 451
CEMT SET PIPELINE 493
ENABLESTATUS option
INQUIRE LIBRARY command 258
ENABLING
CEMT INQUIRE CORBASERVER 161
END
CSFE 615
ENDAFFINITY
CEMT INQUIRE CONNECTION 152
CEMT SET CONNECTION 434
ENDOFDAY
CEMT INQUIRE STATISTICS 303
CEMT SET STATISTICS 499

ENDPOINT
CEMT INQUIRE WEBSERVICE 399
ENQ
CEMT transaction 205
ENQFAILS
CEMT INQUIRE ENQ 206
CEMT INQUIRE UOWENQ 379
ENQMODEL
CEMT DISCARD 124
CEMT INQUIRE ENQMODEL 211
CEMT transaction 209
ENQNAME
CEMT INQUIRE ENQMODEL 211
ENQQUEUE
CEMT SET transaction 463
ENQSCOPE
CEMT INQUIRE ENQ 207
CEMT INQUIRE ENQMODEL 211
CEMT INQUIRE UOWENQ 379
ENQUEUE
CEMT PERFORM STATISTICS 423
EPADAPTER
CEMT DISCARD 124
CEMT INQUIRE transaction 212
CEMT PERFORM STATISTICS 423
CEMT SET transaction 464
EPSTATUS
CEMT INQUIRE EVENTPROCESS 219
EQUAL
CEMT INQUIRE DB2CONN 163, 177
CEMT SET DB2ENTRY 451
ERDSASIZE
CEMT INQUIRE DSAS 194
error message in partition 7
errors in message switching 590
ERRTERM
CMSG 578
ERRTERM INVALID AT XXXXX
CMSG 590
ERRTERM INVLD/UNSUPP XXXX
CMSG 590
ESDSASIZE
CEMT INQUIRE DSAS 194
EUDSASIZE
CEMT INQUIRE DSAS 194
EVENTBINDING
CEMT DISCARD 124
CEMT INQUIRE transaction 216
CEMT PERFORM STATISTICS 423
CEMT SET transaction 465
EVENTPROCESS
CEMT INQUIRE transaction 218
CEMT PERFORM STATISTICS 423
CEMT SET transaction 465
EXCEPT
CEMT INQUIRE MONITOR 264
CEMT SET MONITOR 486
EXCEPTCLASS
CEMT INQUIRE MONITOR 264
EXCI
CEMT INQUIRE CONNECTION 155
CEMT INQUIRE EXCI 220
CEMT INQUIRE transaction 219
EXCLSTATUS
CEMT INQUIRE FILE 226
EXCLUSIVE
CEMT INQUIRE FILE 226
CEMT SET FILE 468
EXECENQ
CEMT INQUIRE ENQ 208
CEMT INQUIRE UOWENQ 380
EXECKEY
CEMT INQUIRE JVM 249
CEMT INQUIRE PROGRAM 289
execution diagnostic facility (EDF) 99
EXECUTIONSET
CEMT INQUIRE PROGRAM 289
EXPIRYINT
CEMT INQUIRE TSMODEL 361
CEMT INQUIRE TSQUEUE 366
EXTENDED
CEMT INQUIRE FILE 228
EXTRA
CEMT INQUIRE TDQUEUE 341

F

F (function) key 114
FACILITY
CEMT INQUIRE TASK 319
FACILITYLIKE
CEMT INQUIRE TRANSACTION 352
FAILED
CEMT INQUIRE CONNECTION 153
CEMT INQUIRE JOURNALNAME 248
CEMT INQUIRE STREAMNAME 306
FAILEDBKOUT
CEMT INQUIRE UOWDSNFAIL 375
Family of resources 115
FCANCEL
CEMT INQUIRE CONNECTION 156
CEMT SET CONNECTION 435
CEMT SET IPCONN 475
FEED
CEMT INQUIRE ATOMSERVICE 129
FENODE option
CEMT DISCARD 124
FEPI
CEMT PERFORM STATISTICS 423
FEPOOL option
CEMT DISCARD 124
FEPROPSET option
CEMT DISCARD 124
FETARGET option
CEMT DISCARD 124
FILE
CEMT DISCARD 124
CEMT INQUIRE ATOMSERVICE 131
CEMT INQUIRE ENQ 208
CEMT INQUIRE FILE 226
CEMT INQUIRE PROCESSTYPE 285
CEMT INQUIRE UOWENQ 380
CEMT PERFORM STATISTICS 423
CEMT SET transaction 466
CEMT transaction 221
FILECOUNT
CEMT INQUIRE DSNAME 200
FILELIMIT
CEMT INQUIRE MONITOR 264
CEMT SET MONITOR 486
fixed-format terminals 537
FLUSH
CEMT INQUIRE JOURNALNAME 248
CEMT SET JOURNALNAME 479
FORCE
CEMT INQUIRE SYSTEM 313
CEMT INQUIRE UOW 371
CEMT SET DSNAME 460
CEMT SET SYSTEM 506
CEMT SET UOW 523
FORCECLOSE
CEMT INQUIRE FILE 228
CEMT INQUIRE VTAM 393
CEMT SET FILE 469
CEMT SET VTAM 526
FORCEDISABLED
CEMT SET DB2ENTRY 451
FORCENOTCON
CEMT SET DB2CONN 440
FORCEPURGE
CEKL SET TASK 109
CEMT INQUIRE CONNECTION 155
CEMT INQUIRE NETNAME 275
CEMT INQUIRE TASK 320
CEMT INQUIRE TERMINAL 347
CEMT PERFORM CLASSCACHE command 409
CEMT PERFORM JVMPOOL command 416
CEMT SET CONNECTION 435
CEMT SET IPCONN 475
CEMT SET JVMPOOL 480
CEMT SET LINE 483
CEMT SET NETNAME 491
CEMT SET TASK 508
CEMT SET TERMINAL 517
CEMT SET WORKREQUEST 530
INQUIRE WORKREQUEST command 403
FORCEQR
CEMT INQUIRE SYSTEM 313
CEMT SET SYSTEM 506
FORCEUOW
CEMT INQUIRE CONNECTION 157
CEMT SET CONNECTION 435
CEMT SET IPCONN 475
FREEING
CEMT INQUIRE CONNECTION 153
CEMT INQUIRE IPCONN 236
FREQUENCY
CEMT INQUIRE MONITOR 264
CEMT SET MONITOR 486
FRLOG
CEMT INQUIRE DSNAME 200
FTYPE
CEMT INQUIRE TASK 319
FULL
CEMT SET PROCESSTYPE 494
FULLAPI
CEMT INQUIRE PROGRAM 289
CEMT SET PROGRAM 496
FULLDATE
CMSG 578

function key 114
FWDRECOVABLE
CEMT INQUIRE DSNAME 201
FWDRECOVLSN
CEMT INQUIRE DSNAME 200

G

GARBAGEINT
CEMT INQUIRE WEB 395
CEMT SET WEB 528
GCMODEL
CEMT INQUIRE JVMSERVER 253
generic names 115
generic resource identifiers 115
GETPAGE
CSPG 619, 622
GMT
CEMT INQUIRE MONITOR 265
GRNAME
CEMT INQUIRE CONNECTION 153
CEMT INQUIRE VTAM 392
GROUP
CEMT INQUIRE DB2ENTRY 174
CEMT SET DB2CONN 440
CEMT SET DB2ENTRY 450
group identifier 542, 547
GROUPID 542, 547
GRPCRITICAL value
INQUIRE TCPIPSERVICE 331
GRSTATUS
CEMT INQUIRE VTAM 392
GTF trace 553
GTFTRACE
CEBT INQUIRE GTFTRACE 80
CEBT INQUIRE transaction 79
CEBT SET transaction 83
CEMT INQUIRE GTFTRACE 230
CEMT INQUIRE transaction 230
CEMT SET transaction 471

H

hardcopy terminal 627
HBACKOUT
CEMT INQUIRE UOW 371
HCOMMIT
CEMT INQUIRE UOW 371
HDISABLED
CEMT INQUIRE URIMAP 388
HEADING
CMSG 578
HEADNG NOT YES/NO AT XXXX
CMSG 590
hexadecimal values 119
HFSFILE
CEMT INQUIRE DOCTEMPLATE 191
CEMT INQUIRE URIMAP 388
HHHHHH
CSPG 619, 626
HHMM
CMSG 578, 590
HHMMSS
CEMT INQUIRE VTAM 393
CEMT SET VTAM 527
HIGH
CEMT INQUIRE DB2CONN 163, 177
CEMT SET DB2ENTRY 451
HOST
CEMT INQUIRE CORBASERVER 161
CEMT INQUIRE IPCONN 237
CEMT INQUIRE transaction 231
CEMT INQUIRE UOWLINK 383
CEMT INQUIRE URIMAP 388
CEMT SET transaction 472
INQUIRE TCPIPSERVICE 332
HOSTCODEPAGE
CEMT INQUIRE URIMAP 389
HOSTTYPE
CEMT INQUIRE CORBASERVER 161
CEMT INQUIRE IPCONN 237
CEMT INQUIRE URIMAP 389
INQUIRE TCPIPSERVICE 332
HTIME
CEMT INQUIRE TASK 319
HTTP
CEMT INQUIRE URIMAP 390
HTTPS
CEMT INQUIRE URIMAP 390
HTYPE
CEMT INQUIRE TASK 319
HVALUE
CEMT INQUIRE TASK 319

I

ID
CMSG 578
ID OPERAND GT 62 CHARS
CMSG 590
ID OPERAND MISSING
CMSG 590
ID PARAM MISSING OR NO)
CMSG 590
IDENTIFIER
CEMT INQUIRE TASK 320
identifiers
resource, lists of 116
identifying terminals 5
IDLE
CEMT INQUIRE DELETSHIPPED 181
CEMT SET DELETSHIPPED 454
IDNTY
CEMT SET MONITOR 487
IDNTYCLASS
CEMT INQUIRE MONITOR 264
IDPROP
CEMT INQUIRE IPCONN 237
IIOP
CEMT INQUIRE UOWLINK 384
CEMT SET WORKREQUEST 530
IMMCLOSE
CEMT INQUIRE IRC 242
CEMT INQUIRE VTAM 393
CEMT SET IRC 478
CEMT SET VTAM 526
IMMEDIATE
CEBT PERFORM SHUTDOWN 86
IMMEDIATE *(continued)*
CEMT PERFORM SHUTDOWN 419
IMMQUIESCED
CEMT INQUIRE DSNAME 201
CEMT SET DSNAME 460
inactivating debugging profiles
with the 3270 interface 42
with the web interface 23
INACTIVE
CEMT INQUIRE SYSTEM 316, 317
INDEXRECFULL
CEMT INQUIRE UOWDSNFAIL 375
INDIRECT
CEMT INQUIRE CONNECTION 151
CEMT INQUIRE TDQUEUE 341
INDOUBT
CEMT INQUIRE TASK 319
CEMT INQUIRE TDQUEUE 339
CEMT INQUIRE TRANSACTION 352
CEMT INQUIRE UOW 371
CEMT INQUIRE UOWDSNFAIL 375
INDOUBTMINS
CEMT INQUIRE TASK 319
CEMT INQUIRE TRANSACTION 353
INDOUBTWAIT
CEMT INQUIRE TASK 320
CEMT INQUIRE TDQUEUE 339
CEMT INQUIRE TRANSACTION 353
INFLIGHT
CEMT INQUIRE UOW 371
Information Display System 577
INITHEAP
CEMT INQUIRE JVMSERVER 253
INPUT
CEMT INQUIRE TDQUEUE 340
CMSG 590
inquire about a resource 114
INQUIRE TCPIPSERVICE 329, 330, 331
INS
CEOT 534
INSERVICE
CEMT INQUIRE CONNECTION 157
CEMT INQUIRE IPCONN 240
CEMT INQUIRE NETNAME 276
CEMT INQUIRE TERMINAL 348
CEMT SET CONNECTION 435
CEMT SET IPCONN 475
CEMT SET LINE 483
CEMT SET NETNAME 491
CEMT SET TERMINAL 517
INTERFACE
CEMT INQUIRE REQUESTMODEL 300
interface to master terminal, programmable 120
INTERNAL
CEMT INQUIRE TASK 321
interregion communication (IRC) 241, 477
INTERVAL
CEMT INQUIRE DELETSHIPPED 182
CEMT INQUIRE STATISTICS 303
CEMT SET DELETSHIPPED 454

INTERVAL *(continued)*
CEMT SET STATISTICS 499
INTFACETYPE
CEMT INQUIRE REQUESTMODEL 300
INTRA
CEMT INQUIRE TDQUEUE 341
INTTRACE
CEBT INQUIRE INTTRACE 80
CEBT INQUIRE transaction 80
CEBT SET transaction 84
CEMT INQUIRE INTTRACE 233
CEMT INQUIRE transaction 232
CEMT SET transaction 473
INVALID
CEMT INQUIRE DSNAME 203
INVALID LDC AT XXXXXXXX
CMSG 590
INVALID OPND AT XXXXX
CMSG 590
INVALID TBL SFX AT XXXXX
CMSG 590
INVLD OPERATR ID AT XXXXX
CMSG 590
INVLD SEPARATOR AT XXXXX
CMSG 590
INVLD TERMINL ID XXXX
CMSG 590
invoking a transaction 1
invoking CEMT from an application program 120
IOERROR
CEMT INQUIRE UOWDSNFAIL 375
IOTYPE
CEMT INQUIRE TDQUEUE 340
IPCONN
CEMT DISCARD 124
CEMT INQUIRE UOWLINK 384
CEMT PERFORM STATISTICS 423
CEMT SET IPCONN 473
IPFAMILY
CEMT INQUIRE CORBASERVER 162
CEMT INQUIRE IPCONN 162, 238, 389
INQUIRE TCPIPSERVICE 332, 333
INQUIRE WORKREQUEST command 403
IPIC
CEMT INQUIRE UOWLINK 383
IPRESOLVED
CEMT INQUIRE CORBASERVER 162
CEMT INQUIRE IPCONN 238
CEMT INQUIRE URIMAP 390
INQUIRE TCPIPSERVICE 333
IRBATCH
CEMT INQUIRE transaction 241
IRC
CEMT INQUIRE CONNECTION 151
CEMT INQUIRE IRC 242
CEMT INQUIRE transaction 241
CEMT INQUIRE UOWLINK 383
CEMT SET transaction 477
IRC (interregion communication)
CEMT INQUIRE transaction 241
CEMT SET transaction 477
ISOLATE
CEMT INQUIRE TRANSACTION 354
ISOLATEST
CEMT INQUIRE TRANSACTION 354

J

JAVA
CEMT INQUIRE PROGRAM 289
JMODEL
CEMT DISCARD 124
CEMT INQUIRE JMODEL 245
CEMT transaction 243
JNDIPREFIX
CEMT INQUIRE CORBASERVER 162
JOURNALNAME
CEMT DISCARD 125
CEMT INQUIRE JMODEL 246
CEMT INQUIRE JOURNALNAME 248
CEMT PERFORM STATISTICS 423
CEMT transaction 246
JVM
CEMT INQUIRE transaction 248
CEMT SET PROGRAM 496
JVMCLASS
CEMT INQUIRE PROGRAM 289
CEMT SET PROGRAM 497
JVMCOUNT
CEMT PERFORM JVMPOOL command 416
JVMPOOL
CEMT INQUIRE transaction 250
CEMT PERFORM STATISTICS 423
CEMT SET transaction 479
CEMT transaction 416
JVMProfile
CEMT SET PROGRAM 497
JVMPROFILE
CEMT INQUIRE JVMSERVER 253
CEMT INQUIRE PROGRAM 289
CEMT PERFORM JVMPOOL command 416
CEMT PERFORM STATISTICS 424
JVMPROGRAM
CEMT PERFORM STATISTICS 424
JVMSERVER
CEMT DISCARD 125
CEMT INQUIRE PROGRAM 289
CEMT INQUIRE transaction 251
CEMT PERFORM STATISTICS 424
CEMT SET transaction 480

K

KEEPTIME
CEMT INQUIRE BRFACILITY 140
key, tab 117
KEYLENGTH
CEMT INQUIRE FILE 226
CEMT SET FILE 469
keyword, minimum abbreviation 3
KILL
CEKL SET TASK 109
CEMT SET IPCONN 475

L

LANGUAGE 542, 547
CEMT INQUIRE PROGRAM 289
Language Environment 289
Language Environment run-time options (CLER) 567
LASTUSEDINT
CEMT INQUIRE TSQUEUE 366
CEMT SET TSQUEUE 523
LCKSTRUCFULL
CEMT INQUIRE UOWDSNFAIL 376
LDC (logical device code) 586
LDC NOT 2 CHARS AT XXXXX
CMSG 590
LENGTH
CEMT INQUIRE PROGRAM 289
CEMT INQUIRE TSQUEUE 366
LENV
CEMT INQUIRE PROGRAM 289
LERUNOPTS
CEMT INQUIRE JVMSERVER 253
LIBRARY
CEMT DISCARD 125
CEMT INQUIRE PROGRAM 289
CEMT PERFORM STATISTICS 424
LIBRARY option
INQUIRE LIBRARY command 257
LIBRARYDSN
CEMT INQUIRE PROGRAM 289
LINE
CEMT INQUIRE LINE 260
CEMT INQUIRE transaction 259
CEMT SET transaction 482
LINK
CEMT INQUIRE UOW 370
CEMT INQUIRE UOWLINK 383
LINKAUTH option
CEMT INQUIRE IPCONN command 238
LINKSYSNET
CEMT INQUIRE BRFACILITY 141
LINKSYSTEM
CEMT INQUIRE BRFACILITY 141
list debugging profiles screen
application debugging profile manager
3270 interface 37
List profiles page
application debugging profile manager
web interface 17
LISTENERPORT option
INQUIRE WORKREQUEST command 403
Lists
of resource identifiers 116
LOAD
CEMT INQUIRE FILE 227
CEMT SET FILE 469
LOADTYPE
CEMT INQUIRE FILE 227

LOCAL
CEMT INQUIRE MONITOR 265
CEMT INQUIRE NETNAME 274
CEMT INQUIRE SYDUMPCODE 310
CEMT INQUIRE TERMINAL 347
CEMT INQUIRE TRDUMPCODE 358
CEMT SET SYDUMPCODE 502
CEMT SET TRDUMPCODE 521
LOCATION
CEMT INQUIRE TSMODEL 362
CEMT INQUIRE TSQUEUE 366
CEMT INQUIRE URIMAP 390
CEMT SET URIMAP 525
LOCKING
CEMT SET FILE 469
LOGDEFER
CEMT INQUIRE SYSTEM 314
CEMT SET SYSTEM 506
logical device code (LDC) 586
logical unit destination 586
LOSTLOCKS
CEMT INQUIRE DSNAME 200
LOW
CEMT INQUIRE DB2CONN 163, 177
CEMT SET DB2ENTRY 451
LSRPOOL
CEMT PERFORM STATISTICS 424
LTPNAME
CEMT INQUIRE PARTNER 278
LU61
CEMT INQUIRE CONNECTION 155
CEMT INQUIRE UOWLINK 383
LUs, logical units
console as a CICS terminal 9

M

MAIN
CEMT INQUIRE TSMODEL 362
CEMT INQUIRE TSQUEUE 367
manager, application debugging profile 15
MANUAL
CEBT SET TAKEOVER 86
Mappinglevel
CEMT INQUIRE WEBSERVICE 400
MAPPINGLEVEL
CEMT INQUIRE XMLTRANSFORM 408
MAPSET
CEMT INQUIRE PROGRAM 289
master terminal operator
duties 6
restriction 6
transaction, CEBT 77
transaction, CEKL 105
transaction, CEMT 113
types of terminal 6
master terminal, programmable interface to 120
MAXACTIVE
CEMT INQUIRE TCLASS 325
CEMT SET TCLASS 509
MAXDATALEN
CEMT SET TCPIPSERVICE 512
INQUIRE TCPIPSERVICE 333
MAXHEAP
CEMT INQUIRE JVMSERVER 253
MAXIMUM
CEMT INQUIRE AUTOINSTALL 134
CEMT INQUIRE MODENAME 262
CEMT INQUIRE SYDUMPCODE 310
CEMT INQUIRE TRDUMPCODE 358
CEMT SET SYDUMPCODE 502
CEMT SET TRDUMPCODE 521
MAXITEMLEN
CEMT INQUIRE TSQUEUE 367
MAXJVMTCBS
CEMT INQUIRE DISPATCHER 183
CEMT SET DISPATCHER 455
MAXNUMRECS
CEMT INQUIRE FILE 227
CEMT SET FILE 469
MAXOPENTCBS
CEMT INQUIRE DISPATCHER 183
CEMT SET DISPATCHER 455
MAXPERSIST
INQUIRE TCPIPSERVICE 333
MAXQTIME
CEMT INQUIRE IPCONN 238
MAXREQS
CEMT SET AUTOINSTALL 428
MAXSOCKETS
CEMT INQUIRE TCPIP 326
CEMT SET TCPIP 510
MAXSSLTCBS
CEMT INQUIRE DISPATCHER 183
CEMT SET DISPATCHER 455
MAXTASKS
CEMT INQUIRE SYSTEM 314
CEMT SET SYSTEM 506
MAXTHRDTCBS
CEMT INQUIRE DISPATCHER 184
MAXXPTCBS
CEMT INQUIRE DISPATCHER 184
CEMT SET DISPATCHER 455
MEDIATYPE
CEMT INQUIRE URIMAP 390
MEMBER
CEMT INQUIRE TDQUEUE 340
MEMBERNAME
CEMT INQUIRE CONNECTION 154
MEMLIMIT
CEMT INQUIRE DSAS 194
CEMT INQUIRE SYSTEM 314
MESSAGE HAS BEEN ROUTED CMSG 590
message-switching
continuation 575
displaying at terminal 622
errors 590
examples 597
message-switching response
indicator 590
response 590
message-switching transaction
multiple destinations 573
message-switching transaction *(continued)*
saved status 575
subsystems 577
terminal systems 577
transaction CMSG 573
messages
+DATE INVLD 2330 to 0030 590
ALL ROUTE ENTRIES DELETED 590
CANCELED - TEMP STRG ERR N 590
canceling 575
chaining 623
chaining in partition 7
CLEAR/CANCL or CONT INPUT/ROUTE/MSG 590
CONTINUE INPUT 590
CONTINUE INPUT OR MSG 590
CONTINUE MSG 590
CONTINUE ROUTE OPERAND 590
DATE ALREADY PASSED 590
DATE INVALID AT xxxxxxxx 590
DATE NOT xx/xx AT xxxxx 590
DATE TOO FAR IN FUTURE 590
delivery 576
displaying 622
ERRTERM INVALID AT xxxxx 590
ERRTERM INVLD/UNSUPP xxxx 590
HEADNG NOT YES/NO AT xxxx 590
ID OPERAND GT 62 CHARS 590
ID PARAM MISSING OR NO) 590
INVALID LDC AT xxxxxxxx 590
INVALID OPND AT xxxxx 590
INVALID TBL SFX AT xxxxx 590
INVLD OPERATR ID AT xxxxx 590
INVLD SEPARATOR AT xxxxx 590
INVLD TERMINL ID xxxx 590
IO OPERAND GT 62 CHARS 590
LDC NOT 2 CHARS AT xxxxx 590
MESSAGE HAS BEEN ROUTED 590
MESSAGE HAS BEEN SENT 629
MSG NOT SENT- NO RECEIVER 590
NEED OPCLASS OR ROUTE 590
NO INPUT - REENTER 590
NO MESSAGE PROVIDED 590
NO MSG STRT QUOT AT xxxxx 590
NO TERM LIST TBL- xx 590
NO TRMID FOR LDC AT xxxxx 590
OPCLASS NOT 1-24 AT xxxxx 590
OPERATORS NOT SIGNDON NNN 590
OPRTR ID NOT SIGNDON /xxx 590
PROTECT OPND NOT 1-2 CHAR 590
purging 7
ROUTE INVALID AT xxxxx 590
sending 573
SPACE MUST FOLLOW TRANSID 590
TERMID GT 4 CHAR AT xxxxx 590
TERMID NOT +/- AT xxxxx 590
TERMINATED BY OPERATOR 590, 629
terminating in partition 7

messages *(continued)*
termination 625
TIME ALREADY PASSED 590
TIME INVALID AT xxxxx 590
TIME NOT 4 CHARS AT xxxxx 590
TOO MANY TBL SFX AT xxxxx 590
UNBALANCD PARENS AT xxxxx 590
UNSUP OPRT TRMNL xxxxxxxx 590
UNSUP TERMINAL xxxxxxxx 590
ZERO LENGTH MSG AT " 590
messages and codes, display 569
MESSCHN
CSPG 623
MINITEMLEN
CEMT INQUIRE TSQUEUE 367
Minrunlevel
CEMT INQUIRE WEBSERVICE 400
MINRUNLEVEL
CEMT INQUIRE XMLTRANSFORM 408
minus sign following CEMT 113
MIRRORLIFE
CEMT INQUIRE IPCONN 239
MM/DD
CMSG 578
MM/DD/YY
CMSG 578
MM/DD/YYYY
CMSG 578
MNPS
CEMT INQUIRE VTAM 393
MOD
CEMT INQUIRE TDQUEUE 338
MODE
CEMT INQUIRE PIPELINE 281
MODENAME
CEMT INQUIRE MODENAME 262
CEMT INQUIRE transaction 261
CEMT SET transaction 483
MODENAME, CEMT requests 483
MODIFY command 9
MODULE
CEMT INQUIRE REQUESTMODEL 301
MONITOR
CEMT INQUIRE MONITOR 264
CEMT INQUIRE transaction 263
CEMT PERFORM STATISTICS 424
CEMT SET transaction 484
monitoring facility transaction CEMN 111
MQCONN
CEMT DISCARD 125
CEMT INQUIRE SYSTEM 314
CEMT INQUIRE transaction 266
CEMT PERFORM STATISTICS 424
CEMT SET MQCONN 488
MQINI
CEMT INQUIRE transaction 269
MQNAME
CEMT INQUIRE MQCONN 266
CEMT SET MQCONN 488
MQQMGR
CEMT INQUIRE MQCONN 266
MQRELEASE
CEMT INQUIRE MQCONN 266
MROBATCH
CEMT INQUIRE DISPATCHER 184
CEMT INQUIRE SYSTEM 314
CEMT SET DISPATCHER 456
CEMT SET SYSTEM 506
MSG
CMSG 578, 590
MSG NOT SENT - NO RECEIVER
CMSG 590
MSGCOPY
CSPG 626
MSGQUEUE1
CEMT INQUIRE DB2CONN 163
CEMT SET DB2CONN 440
MSGQUEUE2
CEMT INQUIRE DB2CONN 163
CEMT SET DB2CONN 440
MSGQUEUE3
CEMT INQUIRE DB2CONN 163
CEMT SET DB2CONN 440
MSGTERM
CSPG 625
MTOMNOXOPST
CEMT INQUIRE PIPELINE 281
MTOMST
CEMT INQUIRE PIPELINE 281
multiple console support 9
multiple destinations, message switching 573
MVS
CEMT INQUIRE JMODEL 246
CEMT INQUIRE JOURNALNAME 248

N

N906D
CEMT INQUIRE DB2CONN 163
NAME
CEMT INQUIRE DOCTEMPLATE 192
NAMEIND
CEMT INQUIRE TDQUEUE 340
NAMESPACE
CEMT INQUIRE BRFACILITY 141
NATLANG, system initialization parameter 542, 548
NATURE
CEMT INQUIRE NETNAME 274
CEMT INQUIRE TERMINAL 346
NEED OPCLASS OR ROUTE
CMSG 590
NET
CEOT 534
NETID
CEMT PERFORM ENDAFFINITY 415
NETNAME
CEMT INQUIRE BRFACILITY 141
CEMT INQUIRE CONNECTION 154
CEMT INQUIRE NETNAME 274
CEMT INQUIRE PARTNER 278
CEMT INQUIRE TERMINAL 347
CEMT INQUIRE transaction 271
CEMT INQUIRE UOW 370
CEMT INQUIRE UOWDSNFAIL 375
NETNAME *(continued)*
CEMT PERFORM ENDAFFINITY 415
CEMT SET transaction 490
NETUOWID
CEMT INQUIRE ENQ 207
CEMT INQUIRE UOW 370
CEMT INQUIRE UOWENQ 379
CEMT INQUIRE UOWLINK 383
NETWORK
CEMT INQUIRE PARTNER 278
networks
CEMT INQUIRE transaction 271
CEMT SET transaction 490
NEWCOPY
CEMT INQUIRE PROGRAM 289
CEMT SET DOCTEMPLATE 457
CEMT SET PROGRAM 497
NEWPS 542, 548
NEXT
CEBT INQUIRE AUXTRACE 78
CEBT SET AUXTRACE 82
CEMT INQUIRE AUXTRACE 136
CEMT SET AUXTRACE 429
NEXTTIME
CEMT INQUIRE STATISTICS 302, 304
CEMT SET STATISTICS 498
NO
CMSG 578
NO INPUT - REENTER
CMSG 590
NO MESSAGE PROVIDED
CMSG 590
NO MSG STRT QUOT AT XXXXX
CMSG 590
NO TERM LIST TBL-XX
CMSG 590
NO TRMID FOR LDC AT XXXXX
CMSG 590
NOADDABLE
CEMT INQUIRE FILE 224
CEMT SET FILE 469
NOANALYZER
CEMT INQUIRE URIMAP 387
NOAPPLNAME
CEMT INQUIRE MONITOR 263
NOATI
CEMT INQUIRE NETNAME 273
CEMT INQUIRE TERMINAL 345
CEMT SET NETNAME 491
CEMT SET TERMINAL 517
CEOT 535
NOAUTHENTIC
CEMT INQUIRE URIMAP 387
NOAUTO
CEMT SET CORBASERVER 440
NOAUTOSWITCH
CEBT INQUIRE DUMP 79
CEBT SET DUMP 83
CEMT INQUIRE DUMPDS 204
CEMT SET DUMPDS 463
NOBROWSE
CEMT INQUIRE FILE 224
CEMT SET FILE 470
NOCEDF
CEMT INQUIRE PROGRAM 289

NOCEDF *(continued)*
CEMT SET PROGRAM 497
NOCMDPROT
CEMT INQUIRE SYSTEM 312
NOCOMPRESS
CEMT INQUIRE MONITOR 263
CEMT SET MONITOR 487
NOCONNECT
CEMT INQUIRE DB2CONN 163
NOCONVERSE
CEMT INQUIRE MONITOR 264
CEMT SET MONITOR 487
NOCREATE
CEMT INQUIRE NETNAME 274
CEMT INQUIRE TERMINAL 346
CEMT SET NETNAME 491
CEMT SET TERMINAL 518
NODAE
CEMT INQUIRE SYDUMPCODE 309
CEMT SET SYDUMPCODE 502
NODEBUG
CEMT SET SYSTEM 506
NODELETE
CEMT INQUIRE FILE 225
CEMT SET FILE 470
NOEMPTYREQ
CEMT INQUIRE FILE 225
CEMT SET FILE 470
NOEXCEPT
CEMT INQUIRE MONITOR 264
CEMT SET MONITOR 487
NOEXCLUSIVE
CEMT INQUIRE FILE 226
CEMT SET FILE 470
NOFORCE
CEMT INQUIRE SYSTEM 313
CEMT SET SYSTEM 506
NOIDNTY
CEMT SET MONITOR 487
NOISOLATE
CEMT INQUIRE TRANSACTION 354
NOLOAD
CEMT INQUIRE FILE 227
CEMT SET FILE 470
NOLOSTLOCKS
CEMT INQUIRE DSNAME 201
NONCRITICAL value
INQUIRE TCPIPSERVICE 331
NONE
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE DB2ENTRY 174
CEMT INQUIRE URIMAP 390
CEMT SET DB2CONN 440
CEMT SET DB2ENTRY 450
CEMT SET URIMAP 525
NONTERMREL
CEMT INQUIRE DB2CONN 163
CEMT SET DB2CONN 440
NOPERF
CEMT INQUIRE MONITOR 264
CEMT SET MONITOR 487
NOPS
CEMT INQUIRE VTAM 393
NOREAD
CEMT INQUIRE FILE 228
NOREAD *(continued)*
CEMT SET FILE 470
NORECOVDATA
CEMT INQUIRE CONNECTION 156
CEMT INQUIRE IPCONN 240
CEMT SET CONNECTION 436
CEMT SET IPCONN 475
NOREENTPROT
CEMT INQUIRE SYSTEM 315
NORELEASE
CEMT INQUIRE DB2CONN 163
CEMT SET DB2CONN 440
NORESRCE
CEMT INQUIRE MONITOR 265
CEMT SET MONITOR 487
NORESTART
CEMT PERFORM SHUTDOWN 419
NORESYNC
CEMT INQUIRE DB2CONN 163
NORETAINED
CEMT INQUIRE DSNAME 202
NORMI
CEMT INQUIRE MONITOR 265
NOROLLBACK
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE DB2ENTRY 176
NOSDTRAN
CEMT PERFORM SHUTDOWN 419
NOSECURITY
CEMT INQUIRE TSMODEL 362
NOSHUTDOWN
CEMT INQUIRE SYDUMPCODE 310
CEMT INQUIRE TRDUMPCODE 358
CEMT SET SYDUMPCODE 503
CEMT SET TRDUMPCODE 521
NOSWITCH
CEBT INQUIRE AUXTRACE 78
CEBT SET AUXTRACE 82
CEMT INQUIRE AUXTRACE 136
CEMT SET AUXTRACE 429
NOSYNCPOINT
CEMT INQUIRE MONITOR 265
CEMT SET MONITOR 487
NOSYSDUMP
CEMT INQUIRE SYDUMPCODE 310
CEMT INQUIRE TRDUMPCODE 359
CEMT SET SYDUMPCODE 503
CEMT SET SYSTEM 506
CEMT SET TRDUMPCODE 521
NOSYSLOG
CEMT INQUIRE STREAMNAME 307
NOTAPPLIC
CEMT INQUIRE ATOMSERVICE 131
CEMT INQUIRE DSNAME 200
CEMT INQUIRE FILE 228
CEMT INQUIRE UOWDSNFAIL 376
notation, syntax 2
NOTCONNECTED
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE MQCONN 266
CEMT SET DB2CONN 440
NOTDEFINED
CEMT INQUIRE PROGRAM 289
NOTDYNAMIC
CEMT INQUIRE PROGRAM 289
NOTEXTENDED
CEMT INQUIRE FILE 228
NOTPENDING
CEMT INQUIRE CONNECTION 154
CEMT INQUIRE IPCONN 239
CEMT SET CONNECTION 436
CEMT SET IPCONN 476
NOTPURGEABLE
CEMT INQUIRE TRANSACTION 354
CEMT SET TRANSACTION 520
NOTRANDUMP
CEMT INQUIRE TRDUMPCODE 359
CEMT SET TRDUMPCODE 521
NOTRECOVABLE
CEMT INQUIRE DSNAME 201
CEMT INQUIRE TSMODEL 362
CEMT INQUIRE TSQUEUE 367
NOTREQUIRED
CEMT INQUIRE PROGRAM 289
NOTRLS
CEMT INQUIRE FILE 229
CEMT INQUIRE UOWDSNFAIL 377
CEMT SET FILE 470
NOTROUTABLE
CEMT INQUIRE TRANSACTION 355
NOTSOS
CEMT INQUIRE DSAS 195, 316
CEMT INQUIRE SYSTEM 316
NOTSUPPORTED
CEMT INQUIRE CONNECTION 153
NOTTABLE
CEMT INQUIRE FILE 229
CEMT SET FILE 470
NOTTI
CEMT INQUIRE NETNAME 277
CEMT INQUIRE TERMINAL 349
CEMT SET NETNAME 491
CEMT SET TERMINAL 518
CEOT 535
NOTWAIT
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE DB2ENTRY 178
CEMT SET DB2ENTRY 452
NOUPDATE
CEMT INQUIRE FILE 229
CEMT SET FILE 470
NOWAIT
CEMT INQUIRE TASK 320
CEMT INQUIRE TDQUEUE 339
CEMT INQUIRE TRANSACTION 353
NQNAME
CEMT INQUIRE CONNECTION 154
CEMT INQUIRE NETNAME 275
CEMT INQUIRE TERMINAL 347
NRS
CEMT INQUIRE CONNECTION 156
CEMT INQUIRE IPCONN 240
NUMBER
CEMT INQUIRE LINE 260

NUMBER *(continued)*
CEMT SET LINE 483
NUMDSNAMES option
INQUIRE LIBRARY command 259
NUMITEMS
CEMT INQUIRE TSQUEUE 367

O

OAPPLID option
INQUIRE WORKREQUEST command 403
OBJECT
CEMT INQUIRE DSNAME 201
OBTAINING
CEMT INQUIRE CONNECTION 153
CEMT INQUIRE IPCONN 236
OCCUPANCY
CEMT INQUIRE JVMSERVER 253
OFF
CEBT SET SURVEILLANCE 85
CEDF 99
CEDX 103
CEMT INQUIRE MONITOR 265
CEMT INQUIRE STATISTICS 305
CEMT SET MONITOR 487
CEMT SET PROCESSTYPE 494
CEMT SET STATISTICS 500
CSFE SPOOLFE 616
CSFE ZCQTRACE 616
OK
CEMT INQUIRE STREAMNAME 306
CEMT INQUIRE UOWLINK 384
OLD
CEMT INQUIRE FILE 225
CEMT INQUIRE TDQUEUE 338
CEMT SET FILE 470
OLDCACHES
CEMT INQUIRE CLASSCACHE 146
ON
CEBT SET SURVEILLANCE 85
CEDF 100
CEDX 103
CEMT INQUIRE MONITOR 265
CEMT INQUIRE STATISTICS 305
CEMT SET MONITOR 487
CEMT SET STATISTICS 500
CSFE SPOOLFE 616
OPCLASS
CMSG 578
OPCLASS NOT 1-24 AT XXXXX
CMSG 590
OPEN
CEBT INQUIRE DUMP 79
CEBT SET DUMP 83
CEMT INQUIRE DUMPDS 204
CEMT INQUIRE FILE 228
CEMT INQUIRE IRC 242
CEMT INQUIRE RRMS 302
CEMT INQUIRE TDQUEUE 340
CEMT INQUIRE VTAM 393
CEMT SET DUMPDS 463
CEMT SET FILE 470
CEMT SET IRC 478
CEMT SET VTAM 527
OPENAPI
CEMT INQUIRE PROGRAM 289
OPENERROR
CEMT INQUIRE UOWDSNFAIL 376
OPENSTATUS
CEMT INQUIRE DUMPDS 204
CEMT INQUIRE FILE 228
CEMT INQUIRE IRC 242
CEMT INQUIRE RRMS 302
CEMT INQUIRE TDQUEUE 340
CEMT INQUIRE VTAM 393
CEMT SET TCPIP 510
CEMT SET TCPIPSERVICE 512
operating system console as a terminal 9
OPERATION
CEMT INQUIRE REQUESTMODEL 301
operator security 4
OPERATORS NOT SIGNDON NNN
CMSG 590
OPID
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE DB2ENTRY 175
CEMT SET DB2CONN 440
CEMT SET DB2ENTRY 450
CMSG 578
OPRTR ID NOT SIGNDON /XXX
CMSG 590
OSLEVEL
CEMT INQUIRE SYSTEM 314
OTASK option
INQUIRE WORKREQUEST command 403
OTRANSID option
INQUIRE WORKREQUEST command 403
OTSTID
CEMT INQUIRE UOW 370
OTSTIMEOUT
CEMT INQUIRE TRANSACTION 354
OUTPUT
CEMT INQUIRE TDQUEUE 340
OUTSERVICE
CEMT INQUIRE CONNECTION 157
CEMT INQUIRE IPCONN 240
CEMT INQUIRE NETNAME 276
CEMT INQUIRE TERMINAL 348
CEMT SET CONNECTION 437
CEMT SET IPCONN 476
CEMT SET LINE 483
CEMT SET NETNAME 492
CEMT SET TERMINAL 518
overtyping a display 117
OWNER
CEMT INQUIRE ENQ 207
CEMT INQUIRE UOWENQ 380

P

PA1 print key 7
PAG
CEOT 534
page copying 7, 626
page retrieval
and partitions 7, 622
page retrieval *(continued)*
logical device 621
session 619
single keystroke 621
transaction, CSPG 619
page retrieval function
new PROFILE definition, DFHCICSP 3
PAGEABLE
CEMT INQUIRE NETNAME 275
CEMT INQUIRE TERMINAL 347
CEMT SET NETNAME 492
CEMT SET TERMINAL 518
CEOT 535
PAGESTATUS
CEMT INQUIRE NETNAME 275
CEMT INQUIRE TERMINAL 347
paging command 619
PARAMETERS
CSPG 623
PARTCOUNT
CEMT INQUIRE BUNDLE 144
partitions 622
partitions, BMS 7
PARTITIONSET
CEMT INQUIRE PROGRAM 289
PARTNER
CEMT DISCARD 125
CEMT INQUIRE IPCONN 239
CEMT INQUIRE PARTNER 278
CEMT INQUIRE transaction 277
password phrases
and the CESN transaction 548
PATH
CEMT INQUIRE DSNAME 201
CEMT INQUIRE URIMAP 390
PAUSE
CEBT INQUIRE AUXTRACE 78
CEBT SET AUXTRACE 82
CEMT INQUIRE AUXTRACE 136
CEMT SET AUXTRACE 429
PENDING
CEMT INQUIRE CONNECTION 154
CEMT INQUIRE IPCONN 239
PENDSTATUS
CEMT INQUIRE CONNECTION 154
CEMT INQUIRE IPCONN 239
PERF
CEMT INQUIRE MONITOR 264
CEMT SET MONITOR 487
PERFCLASS
CEMT INQUIRE MONITOR 264
PERFORM, CEBT
SHUTDOWN 86
SNAP 86
TAKEOVER 87
PERFORM, CEMT
RESET 418
SECURITY 418
SHUTDOWN 419
STATISTICS 420
PERMANENT
CEMT INQUIRE URIMAP 390
CEMT SET URIMAP 525
PGCHAIN, system initialization parameter 619, 623

PGCOPY, system initialization
parameter 619, 626
PGMINTERFACE
CEMT INQUIRE WEBSERVICE 400
PGPURGE, system initialization
parameter 619, 625
PGRET, system initialization
parameter 619, 622
PHASEIN
CEMT INQUIRE PROGRAM 289
CEMT SET PROGRAM 497
PHASEOUT
CEMT PERFORM CLASSCACHE
command 410
CEMT PERFORM JVMPOOL
command 417
CEMT SET JVMPOOL 480
PHASINGOUT
CEMT INQUIRE CLASSCACHE 146
CEMT INQUIRE JVMPOOL 250
PHASINGOUTST
CEMT INQUIRE JVM 249
PID
CEMT INQUIRE JVMSERVER 253
PIPELINE
CEMT DISCARD 125
CEMT INQUIRE URIMAP 391
CEMT INQUIRE WEBSERVICE 400
CEMT PERFORM PIPELINE 418
CEMT PERFORM STATISTICS 424
CEMT SET PIPELINE 492
CEMT transaction 417
PLAN
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE DB2ENTRY 177
CEMT SET DB2CONN 440
CEMT SET DB2ENTRY 451
PLANEXITNAME
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE DB2ENTRY 177
CEMT SET DB2CONN 440
CEMT SET DB2ENTRY 451
CONNECT
CEMT SET DB2CONN 440
EQUAL
CEMT SET DB2CONN 440
HIGH
CEMT SET DB2CONN 440
LOW
CEMT SET DB2CONN 440
NOCONNECT
CEMT SET DB2CONN 440
NORESYNC
CEMT SET DB2CONN 440
NOTWAIT
CEMT SET DB2CONN 440
PRIORITY
CEMT SET DB2CONN 440
PURGECYCLEM
CEMT SET DB2CONN 440
PURGECYCLES
CEMT SET DB2CONN 440
REBUILD
CEMT SET DB2CONN 440
RECONNECT
CEMT SET DB2CONN 440
PLANEXITNAME *(continued)*
RESYNC
CEMT SET DB2CONN 440
RESYNCMEMBER
CEMT SET DB2CONN 440
REUSELIMIT
CEMT SET DB2CONN 440
SECURITY
CEMT SET DB2CONN 440
SIGNID
CEMT SET DB2CONN 440
STANDBYMODE
CEMT SET DB2CONN 440
STATSQUEUE
CEMT SET DB2CONN 440
TCBLIMIT
CEMT SET DB2CONN 440
THREADLIMIT
CEMT SET DB2CONN 440
THREADWAIT
CEMT SET DB2CONN 440
TWAIT
CEMT SET DB2CONN 440
PLi
CEMT INQUIRE PROGRAM 289
PLT
CEMT PERFORM SHUTDOWN 419
plus sign in scrolling under CEMT 117
POOL
CEMT SET DB2ENTRY 451
POOLNAME
CEMT INQUIRE TSMODEL 362
CEMT INQUIRE TSQUEUE 365
CEMT SET TSQUEUE 523
PORT
CEMT INQUIRE IPCONN 240
CEMT INQUIRE URIMAP 390
positional fields 117
PREFIX
CEMT INQUIRE TSMODEL 362
CMSG 578
PRFILE
CEMT INQUIRE
TRANSACTION 354
PRGDLAY, system initialization
parameter 578
PRI
CEOT 534
PRINT
CSFE 615
print (PA1) key 7
PRIORITY
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE DB2ENTRY 177
CEMT INQUIRE NETNAME 275
CEMT INQUIRE TASK 320
CEMT INQUIRE TERMINAL 347
CEMT INQUIRE
TRANSACTION 354
CEMT SET DB2ENTRY 451
CEMT SET NETNAME 492
CEMT SET TASK 508
CEMT SET TERMINAL 518
CEMT SET TRANSACTION 520
PRIVATE
CEMT INQUIRE PROGRAM 289
CEMT SET PROGRAM 498
PROCESS
CEMT INQUIRE TASK 320
CEMT SET PROCESSTYPE 494
PROCESSTYPE
CEMT DISCARD 125
CEMT INQUIRE TASK 320
CEMT SET transaction 494
PROFILE
CEMT DISCARD 125
CEMT INQUIRE JVM 250
CEMT INQUIRE PARTNER 278
CEMT INQUIRE PROFILE 288
CEMT INQUIRE transaction 286
profile manager, application
debugging 15
profile manager, debugging
web interface 15
configuring 15
profile, rebuild 418
PROGAUTO
CEMT PERFORM STATISTICS 424
PROGAUTOCTLG
CEMT INQUIRE SYSTEM 314
CEMT SET SYSTEM 505
PROGAUTOEXIT
CEMT INQUIRE SYSTEM 315
CEMT SET SYSTEM 506
PROGAUTOINST
CEMT INQUIRE SYSTEM 315
CEMT SET SYSTEM 504
PROGRAM
CEMT DISCARD 125
CEMT INQUIRE
ATOMSERVICE 131
CEMT INQUIRE
AUTOINSTALL 134
CEMT INQUIRE PROGRAM 289
CEMT INQUIRE transaction 289
CEMT INQUIRE
TRANSACTION 354
CEMT INQUIRE URIMAP 390
CEMT INQUIRE WEBSERVICE 400
CEMT PERFORM STATISTICS 424
CEMT SET AUTOINSTALL 428
CEMT SET PROGRAM 495
CEMT SET transaction 495
PROGRAMDEF
CEMT PERFORM STATISTICS 424
programmable interface to master
terminal 120
PROGTYPE
CEMT INQUIRE PROGRAM 289
PROTECT
CMSG 578
PROTECT OPND NOT 1-2 CHAR
CMSG 590
PROTECTNUM
CEMT INQUIRE DB2ENTRY 177
CEMT SET DB2ENTRY 452
PROTOCOL
CEMT INQUIRE CONNECTION 155
CEMT INQUIRE UOWLINK 383
PS
CESN 542, 548
PSDINTERVAL
CEMT INQUIRE VTAM 393
CEMT SET VTAM 527

pseudo-display fields 117
PSTYPE
CEMT INQUIRE VTAM 393
PTHREADS
CEMT INQUIRE DB2ENTRY 177
PUBLISH
CEMT PERFORM CORBASERVER 412
CEMT PERFORM DJAR 414
PURGE
CEKL SET TASK 109
CEMT INQUIRE CONNECTION 155
CEMT INQUIRE NETNAME 275
CEMT INQUIRE TASK 320
CEMT INQUIRE TERMINAL 347
CEMT PERFORM CLASSCACHE command 410
CEMT PERFORM JVMPOOL command 417
CEMT SET CONNECTION 438
CEMT SET IPCONN 476
CEMT SET JVMPOOL 480
CEMT SET LINE 483
CEMT SET NETNAME 492
CEMT SET TASK 508
CEMT SET TERMINAL 518
CEMT SET WORKREQUEST 530
INQUIRE WORKREQUEST command 404
PURGEABILITY
CEMT INQUIRE TRANSACTION 354
PURGEABLE
CEMT INQUIRE TRANSACTION 354
CEMT SET TRANSACTION 520
PURGECYCLEM
CEMT INQUIRE DB2CONN 163
PURGECYCLES
CEMT INQUIRE DB2CONN 163
PURGETHRESH
CEMT INQUIRE TCLASS 325
CEMT SET TCLASS 509
PURGETYPE
CEMT INQUIRE CONNECTION 155
CEMT INQUIRE NETNAME 275
CEMT INQUIRE TASK 320
CEMT INQUIRE TERMINAL 347
PURGETYPE option
INQUIRE WORKREQUEST command 403

Q

Q
QCSPG 622
QD
CEMT INQUIRE TASK 321
QLEN
CEMT INQUIRE ENQ 207
CEMT INQUIRE UOWENQ 379
QR
CEMT INQUIRE TASK 321
QUALIFIER
CEMT INQUIRE ENQ 207
CEMT INQUIRE UOWENQ 379
QUASIRENT
CEMT INQUIRE PROGRAM 289
question mark preceding CEMT 113
QUEUE
CEMT INQUIRE TDQUEUE 340
CEMT SET transaction 498
QUEUED
CEMT INQUIRE TCLASS 325
QUEUELIMIT
CEMT INQUIRE IPCONN 240
queues, CEMT requests 335, 498, 509
QUIESCED
CEMT INQUIRE DSNAME 201
CEMT SET DSNAME 460
QUIESCESTATE
CEMT INQUIRE DSNAME 201

R

RACF resource profile 418
RANKING option
INQUIRE LIBRARY command 259
RBATYPE
CEMT INQUIRE FILE 228
RCONNECTION
CEMT INQUIRE CONNECTION 151
RDBACK
CEMT INQUIRE TDQUEUE 340
RDO (resource definition online) 91
RDSASIZE
CEMT INQUIRE DSAS 194
READ
CEMT INQUIRE FILE 228
CEMT SET FILE 470
READSTATUS
CEMT INQUIRE FILE 228
REASON
CEMT INQUIRE UOWDSNFAIL 375
REBUILD
CEMT PERFORM SECURITY 419
rebuild resource profiles 418
RECEIVECOUNT
CEMT INQUIRE IPCONN 240
RECONNECT
CEMT INQUIRE DB2CONN 163
RECORD
CEMT INQUIRE STATISTICS 304
CEMT PERFORM STATISTICS 424
RECORDING
CEMT INQUIRE STATISTICS 305
RECORDNOW
CEMT SET STATISTICS 500
RECORDSIZE
CEMT INQUIRE FILE 228
CEMT SET FILE 470
RECOVDATA
CEMT INQUIRE CONNECTION 156
CEMT INQUIRE IPCONN 240
RECOVERABLE
CEMT INQUIRE DSNAME 202
CEMT INQUIRE TSMODEL 362
CEMT INQUIRE TSQUEUE 367
RECOVERED
CEMT INQUIRE DSNAME 198
CEMT SET DSNAME 461
RECOVERLOCKS
CEMT INQUIRE DSNAME 201
RECOVERY
CEMT PERFORM STATISTICS 424
RECOVSTATUS
CEMT INQUIRE CONNECTION 156
CEMT INQUIRE DSNAME 201
CEMT INQUIRE IPCONN 240
CEMT INQUIRE TSMODEL 362
CEMT INQUIRE TSQUEUE 367
REDIRECTTYPE
CEMT INQUIRE URIMAP 390
REENTPROT
CEMT INQUIRE SYSTEM 315
REENTPROTECT
CEMT INQUIRE SYSTEM 315
REGERROR
INQUIRE TCPIPSERVICE 331
REGISTERED
CEMT INQUIRE VTAM 392
INQUIRE TCPIPSERVICE 331
REJECT
CEMT INQUIRE TDQUEUE 340
RELATED
CEMT INQUIRE SYDUMPCODE 310
CEMT INQUIRE TRDUMPCODE 358
CEMT SET SYDUMPCODE 503
CEMT SET TRDUMPCODE 521
RELATION
CEMT INQUIRE ENQ 207
CEMT INQUIRE UOWENQ 380
RELEASE
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE SYSTEM 315
CEMT SET DB2CONN 440
RELEASED
CEMT INQUIRE CONNECTION 153
CEMT INQUIRE IPCONN 236
CEMT INQUIRE NETNAME 276
CEMT INQUIRE TERMINAL 348
CEMT SET BRFACILITY 430
CEMT SET CONNECTION 438
CEMT SET IPCONN 477
CEMT SET NETNAME 492
CEMT SET TERMINAL 518
RELOAD
CEMT PERFORM CLASSCACHE command 410
REMLOSTLOCKS
CEMT INQUIRE DSNAME 201
REMOTE
CEMT INQUIRE FILE 223
CEMT INQUIRE TDQUEUE 341
REMOTEPREFIX
CEMT INQUIRE TSMODEL 362
REMOTESYSNET
CEMT INQUIRE BRFACILITY 141
REMOTESYSTEM
CEMT INQUIRE BRFACILITY 141
CEMT INQUIRE CONNECTION 157
CEMT INQUIRE NETNAME 275
CEMT INQUIRE PROGRAM 289
CEMT INQUIRE TERMINAL 347
CEMT INQUIRE TSMODEL 362
REMOVE
CEMT INQUIRE DSNAME 197

REMOVE *(continued)*
CEMT INQUIRE SYDUMPCODE 309
CEMT INQUIRE TRDUMPCODE 358
CEMT SET DSNAME 461
CEMT SET SYDUMPCODE 503
CEMT SET TRDUMPCODE 521
REPLY command 10
REQUESTID option
INQUIRE WORKREQUEST command 404
REQUESTMODEL
CEMT DISCARD 126
CEMT PERFORM STATISTICS 424
CEMT transaction 297
REQUIRED
CEMT INQUIRE PROGRAM 289
RESCOUNT
CEMT INQUIRE PROGRAM 289
RESET
CEMT INQUIRE JOURNALNAME 248
CEMT INQUIRE STATISTICS 305
CEMT INQUIRE SYDUMPCODE 309
CEMT INQUIRE TRDUMPCODE 358
CEMT PERFORM RESET 418
CEMT SET JOURNALNAME 479
CEMT SET SYDUMPCODE 503
CEMT SET TRDUMPCODE 522
RESETLOCKS
CEMT INQUIRE DSNAME 198
CEMT SET DSNAME 461
RESETNOW
CEMT PERFORM STATISTICS 424
CEMT SET STATISTICS 500
RESOURCE
CEMT INQUIRE ENQ 207
CEMT INQUIRE UOWENQ 380
resource definition online transaction, CEDA 91
resource definitions
profile, DFHCICSP 3
resource groups
Subdefining 116
resource identifiers
lists of 116
RESOURCENAME
CEMT INQUIRE ATOMSERVICE 131
RESOURCETYPE
CEMT INQUIRE ATOMSERVICE 131
RESPWAIT
CEMT INQUIRE PIPELINE 281
CEMT SET PIPELINE 493
RESRCE
CEMT INQUIRE MONITOR 265
CEMT SET MONITOR 487
RESRCECLASS
CEMT INQUIRE MONITOR 264
RESYNC
CEMT INQUIRE CONNECTION 157
CEMT INQUIRE DB2CONN 163
CEMT SET CONNECTION 438
RESYNC *(continued)*
CEMT SET IPCONN 477
RESYNCMEMBER
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE MQCONN 266
CEMT SET MQCONN 488
RESYNCSTATUS
CEMT INQUIRE UOWLINK 383
RETAINED
CEMT INQUIRE DSNAME 202
CEMT INQUIRE ENQ 207
CEMT INQUIRE UOWENQ 380
RETLOCKS
CEMT INQUIRE DSNAME 202
RETRACT
CEMT PERFORM CORBASERVER 412
CEMT PERFORM DJAR 414
retrieve a page 7, 619
RETRY
CEMT INQUIRE DSNAME 199
CEMT SET DSNAME 462
REUSELIMIT
CEMT INQUIRE DB2CONN 163
REUSEST
CEMT INQUIRE CLASSCACHE 146
CEMT INQUIRE JVM 250
RLEN
CEMT INQUIRE ENQ 207
CEMT INQUIRE UOWENQ 380
rls
CEMT INQUIRE UOWDSNFAIL 377
RLS
CEMT INQUIRE FILE 229
CEMT SET FILE 470
RLSACCESS
CEMT INQUIRE FILE 229
CEMT INQUIRE UOWDSNFAIL 377
RLSGONE
CEMT INQUIRE UOWDSNFAIL 376
RLSSERVER
CEMT INQUIRE UOW 372
CEMT INQUIRE UOWDSNFAIL 374
RMI
CEMT INQUIRE MONITOR 265
CEMT INQUIRE UOWLINK 384
RMIQFY
CEMT INQUIRE UOWLINK 384
RMIST
CEMT INQUIRE MONITOR 265
RNAME 157
CEMT INQUIRE NETNAME 275
CEMT INQUIRE TERMINAL 348
RNETNAME 157
CEMT INQUIRE NETNAME 276
CEMT INQUIRE TERMINAL 348
ROLE
CEMT INQUIRE UOWLINK 384
ROLLBACK
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE DB2ENTRY 176
ROUTABLE
CEMT INQUIRE TRANSACTION 355
ROUTE
CMSG 578, 590
ROUTE INVALID AT XXXXX
CMSG 590
routed pages 627
routing transaction, CRTE 613
ROUTSTATUS
CEMT INQUIRE TRANSACTION 355
RRCOMMITFAIL
CEMT INQUIRE UOWDSNFAIL 376
RRINDOUBT
CEMT INQUIRE UOWDSNFAIL 376
RRMS
CEMT INQUIRE transaction 301
CEMT INQUIRE UOW 372
CEMT INQUIRE UOWLINK 383
RSESSION
CEMT INQUIRE NETNAME 274
CEMT INQUIRE TERMINAL 346
RTERMINAL
CEMT INQUIRE NETNAME 274
CEMT INQUIRE TERMINAL 346
RUNAWAY
CEMT INQUIRE DISPATCHER 184
CEMT INQUIRE SYSTEM 315
CEMT SET DISPATCHER 456
CEMT SET SYSTEM 507
RUNNING
CEKL INQUIRE TASK 106
CEMT INQUIRE TASK 321
running CIND 562
RUNSTATUS
CEMT INQUIRE TASK 321
RUNTIME
CEMT INQUIRE PROGRAM 289

S

S
CEMT INQUIRE TASK 321
saved status, message switching 575
SCAN
CEMT PERFORM CORBASERVER 412
CEMT PERFORM PIPELINE 418
SCANDELAY
CEMT INQUIRE DISPATCHER 184
CEMT INQUIRE SYSTEM 316
CEMT SET DISPATCHER 456
CEMT SET SYSTEM 507
SCHEME
CEMT INQUIRE URIMAP 390
scrolling a display 117
SD
CEMT INQUIRE TASK 321
SDSASIZE
CEMT INQUIRE DSAS 194
SDTRAN
CEMT INQUIRE SYSTEM 316
CEMT PERFORM SHUTDOWN 420
SEARCHPOS option
INQUIRE LIBRARY command 259
SECURITY 418
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE TSMODEL 362
security checking 96, 120
security, operator 4

SECURITYNAME
CEMT INQUIRE IPCONN 240
SECURITYST
CEMT INQUIRE TSMODEL 362
SEND
CMSG 578
SENDCOUNT
CEMT INQUIRE IPCONN 240
SENDMTOMST
CEMT INQUIRE PIPELINE 282
sequential terminal 577
SERVER
CEMT INQUIRE URIMAP 391
SERVICE
CEMT INQUIRE ATOMSERVICE 129
SERVSTATUS
CEMT INQUIRE CONNECTION 157
CEMT INQUIRE IPCONN 240
CEMT INQUIRE LINE 260
CEMT INQUIRE NETNAME 276
CEMT INQUIRE TERMINAL 348
SESSBEANTIME
CEMT INQUIRE CORBASERVER 163
CEMT SET CORBASERVER 440
SESSION
CEMT INQUIRE NETNAME 274
CEMT INQUIRE TERMINAL 346
SESSIONID
CEDF 100
set LE debugging display device screen
application debugging profile manager
3270 interface 56
Set LE display device page
application debugging profile manager
web interface 35
setting the display device
with the 3270 interface 55
with the web interface 34
SHARE
CEMT INQUIRE FILE 225
CEMT SET FILE 471
SHARED
CEMT INQUIRE PROGRAM 289
CEMT SET PROGRAM 498
SHARESTATUS
CEMT INQUIRE PROGRAM 289
SHELF
CEMT INQUIRE CORBASERVER 163
CEMT INQUIRE PIPELINE 282
shipped terminal definitions
deletion of
CEMT DJAR command 413
CEMT INQUIRE DELETSHIPPED 181
CEMT PERFORM CORBASERVER command 411
CEMT PERFORM DELETSHIPPED command 413
CEMT SET DELETSHIPPED command 453
SHR
CEMT INQUIRE TDQUEUE 338
SHUNTED
CEMT INQUIRE UOW 372
SHUTDOWN
CEMT INQUIRE SYDUMPCODE 310
CEMT INQUIRE TRDUMPCODE 358
CEMT SET SYDUMPCODE 503
CEMT SET TRDUMPCODE 522
shutdown, CICS
CEBT 86
CEMT 419
SHUTOPTION
CEMT INQUIRE SYDUMPCODE 310
CEMT INQUIRE TRDUMPCODE 358
SIGN
CEMT INQUIRE DB2ENTRY 175
CEMT SET DB2CONN 440
CEMT SET DB2ENTRY 450
sign-off, CESF
fixed-format terminal 537
SIGNID
CEMT INQUIRE DB2CONN 163
signon, CESL
3270 screen 540
non-3270 screen 541
signon, CESN
3270 screen 546
non-3270 screen 547
single keystroke retrieval (SKR) 621
SIZE
CEMT INQUIRE DOCTEMPLATE 192
SKR (single keystroke retrieval) 621
SKRxxxx, system initialization parameter 621
SMF
CEMT INQUIRE JMODEL 246
CEMT INQUIRE JOURNALNAME 248
SNAP
CEMT PERFORM DUMP 414
CEMT transaction 414
SNAP command
CEBT transaction 86
SNPS
CEMT INQUIRE VTAM 393
SOAP
CEMT SET WORKREQUEST 530
SOAPlevel
CEMT INQUIRE PIPELINE 282
SOCKETCLOSE option
INQUIRE URIMAP command 390
SOCKPOOLSIZE option
INQUIRE URIMAP command 391
SOS
CEMT INQUIRE DSAS 195, 316
CEMT INQUIRE SYSTEM 316
SOSABOVEBAR
CEMT INQUIRE DSAS 194, 316
SOSABOVELINE
CEMT INQUIRE DSAS 195
CEMT INQUIRE SYSTEM 316
SOSBELOWLINE
CEMT INQUIRE DSAS 195
SOSBELOWLINE *(continued)*
CEMT INQUIRE SYSTEM 316
SPACE MUST FOLLOW TRANSID
CMSG 590
SPOOLFE
CSFE 616
spooling interface trace 616
SQLCODE
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE DB2ENTRY 176
CEMT SET DB2CONN 440
CEMT SET DB2ENTRY 451
SSLTYPE option
CEMT INQUIRE IPCONN command 240
SSLUNAUTH
CEMT INQUIRE CORBASERVER 163
STACK option
INQUIRE WORKREQUEST command 404
STANDBYMODE
CEMT INQUIRE DB2CONN 163
START
CEBT INQUIRE AUXTRACE 78
CEBT INQUIRE GTFTRACE 80
CEBT INQUIRE INTTRACE 80
CEBT SET AUXTRACE 82
CEBT SET GTFTRACE 84
CEBT SET INTTRACE 84
CEMT INQUIRE AUXTRACE 136
CEMT INQUIRE GTFTRACE 231
CEMT INQUIRE INTTRACE 233
CEMT PERFORM CLASSCACHE command 410
CEMT PERFORM JVMPOOL 417
CEMT SET AUXTRACE 429
CEMT SET EVENTPROCESS 466
CEMT SET GTFTRACE 471
CEMT SET INTTRACE 473
STARTCODE
CEMT INQUIRE TASK 321
STARTING
CEMT INQUIRE UOWLINK 384
starting a transaction 1
STARTTIME
CEMT INQUIRE CLASSCACHE 146
STATE
CEMT INQUIRE DJAR 188
CEMT INQUIRE ENQ 207
CEMT INQUIRE UOWENQ 380
STATIC
CEMT INQUIRE DSNAME 199
STATISTICS
CEMT INQUIRE STATISTICS 305
CEMT INQUIRE transaction 302
CEMT PERFORM 420
CEMT SET transaction 498
STATS
CEMT PERFORM STATISTICS 424
STATSQUEUE
CEMT INQUIRE DB2CONN 163
status
hardcopy terminal 627
indicator 590
STATUS
CEMT INQUIRE AUXTRACE 136

STATUS *(continued)*
CEMT INQUIRE CLASSCACHE 146
CEMT INQUIRE ENQMODEL 212
CEMT INQUIRE GTFTRACE 230, 233
CEMT INQUIRE JOURNALNAME 248
CEMT INQUIRE JVMPOOL 250
CEMT INQUIRE JVMSERVER 253
CEMT INQUIRE MONITOR 265
CEMT INQUIRE PROGRAM 289
CEMT INQUIRE STREAMNAME 306
CEMT INQUIRE TRANSACTION 355
STOP
CEBT INQUIRE AUXTRACE 78
CEBT SET AUXTRACE 82
CEBT SET GTFTRACE 84
CEBT SET INTTRACE 85
CEMT INQUIRE AUXTRACE 136
CEMT INQUIRE GTFTRACE 231
CEMT INQUIRE INTTRACE 233
CEMT SET AUXTRACE 429
CEMT SET EVENTPROCESS 466
CEMT SET GTFTRACE 471
CEMT SET INTTRACE 473
STOPPED
CEBT INQUIRE GTFTRACE 80
CEBT INQUIRE INTTRACE 80
STORAGE
CEMT PERFORM STATISTICS 424
storage subpool 617
STOREPROTECT
CEMT INQUIRE SYSTEM 316
STREAMNAME
CEMT INQUIRE JMODEL 246
CEMT INQUIRE JOURNALNAME 248
CEMT INQUIRE STREAMNAME 306
CEMT PERFORM STATISTICS 424
CEMT transaction 305
Subdefining
resource groups 116
SUBORDINATE
CEMT INQUIRE UOWLINK 384
subsystem message-switching 577
SUBTASKS
CEMT INQUIRE DISPATCHER 184
supervisory terminal operator
duties 5
identifying terminals 5
SUPRID option 5
transaction CEST 551
SURVEILLANCE
CEBT transaction 81, 85
SUSPENDED
CEKL INQUIRE TASK 106
CEMT INQUIRE TASK 321
SWITCH
CEBT INQUIRE AUXTRACE 78
CEBT INQUIRE DUMP 79
CEBT SET AUXTRACE 82
CEBT SET DUMP 83
CEMT INQUIRE AUXTRACE 136
CEMT INQUIRE DUMPDS 204
SWITCH *(continued)*
CEMT SET AUXTRACE 429
CEMT SET DUMPDS 463
SWITCHACTION
CEMT INQUIRE AUXTRACE 136
SWITCHSTATUS
CEMT INQUIRE AUXTRACE 136
CEMT INQUIRE DUMPDS 204
SYDUMPCODE
CEMT INQUIRE SYDUMPCODE 310
CEMT SET transaction 501
CEMT transaction 307
SYNCPOINT
CEMT INQUIRE MONITOR 265
CEMT SET MONITOR 487
SYNCPOINTST
CEMT INQUIRE MONITOR 265
syntax notation 2
SYSDUMP
CEMT INQUIRE SYDUMPCODE 310
CEMT INQUIRE TRDUMPCODE 358
CEMT PERFORM STATISTICS 424
CEMT SET SYDUMPCODE 503
CEMT SET SYSTEM 507
CEMT SET TRDUMPCODE 522
SYSDUMPING
CEMT INQUIRE SYDUMPCODE 310
CEMT INQUIRE TRDUMPCODE 358
SYSID
CEDF 100
CEMT INQUIRE TSQUEUE 364
CEMT INQUIRE UOW 370
CEMT INQUIRE UOWDSNFAIL 377
CEMT INQUIRE UOWLINK 384
CEMT SET TSQUEUE 523
CRTE transaction 613
SYSLOG
CEMT INQUIRE STREAMNAME 307
SYSTEM
CEMT INQUIRE transaction 310
CEMT SET transaction 503
system spooling interface trace 616
SYSTEMLOG
CEMT INQUIRE STREAMNAME 307

T

tab key 117
TABLE
CEMT INQUIRE FILE 229
TABLEMGR
CEMT PERFORM STATISTICS 424
TABLENAME
CEMT INQUIRE FILE 229
CEMT SET FILE 471
TAKEOVER
CEBT 81
CEBT transaction 85, 87
CEMT PERFORM SHUTDOWN 420
TARGETCOUNT
CEMT INQUIRE BUNDLE 144
TARGETSYS option
INQUIRE WORKREQUEST command 404
task
CEKL INQUIRE TASK 106
Task
CEKL SET TASK 109
TASK
CEKL SET transaction 108
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE DB2ENTRY 174
CEMT INQUIRE EXCI 220
CEMT INQUIRE JVM 250
CEMT INQUIRE NETNAME 276
CEMT INQUIRE TASK 319, 321
CEMT INQUIRE TERMINAL 348
CEMT SET DB2CONN 440
CEMT SET DB2ENTRY 449
CEMT SET transaction 507
TASK command
CEKL INQUIRE transaction 105
CEMT INQUIRE transaction 317
TASK option
INQUIRE WORKREQUEST command 404
TASKDATAKEY
CEMT INQUIRE TRANSACTION 355
TASKDATALOC
CEMT INQUIRE TRANSACTION 355
TASKID
CEMT INQUIRE BRFACILITY 141
CEMT INQUIRE ENQ 207
CEMT INQUIRE UOW 370
CEMT INQUIRE UOWENQ 380
tasks
CEKL INQUIRE requests 105
CEKL SET requests 108
CEMT INQUIRE requests 317
CEMT SET requests 507
TASKS
CEMT INQUIRE MQCONN 266
TCB
CEMT INQUIRE TASK 321
TCBLIMIT
CEMT INQUIRE DB2CONN 163
TCBs
CEMT INQUIRE DB2CONN 163
TCLASS
CEMT DISCARD 126
CEMT INQUIRE TASK 317
CEMT INQUIRE TCLASS 325
CEMT INQUIRE transaction 322
CEMT INQUIRE TRANSACTION 356
CEMT PERFORM STATISTICS 424
CEMT SET transaction 509
CEMT SET TRANSACTION 520
TCP/IP services, CEMT requests 325, 327
TCPIP
CEMT INQUIRE transaction 325
CEMT PERFORM STATISTICS 425
CEMT SET transaction 509

TCPIP, CEMT requests 511
TCPipservice
  CEMT PERFORM STATISTICS 425
TCPIPSERVICE
  CEMT DISCARD 126
  CEMT INQUIRE HOST 232
  CEMT INQUIRE IPCONN 241
  CEMT INQUIRE transaction 327
  CEMT INQUIRE URIMAP 391
  CEMT SET HOST 472
  CEMT SET transaction 511
TDQ
  CEMT INQUIRE ENQ 208
  CEMT INQUIRE UOWENQ 381
TDQUEUE
  CEMT DISCARD 126
  CEMT INQUIRE TDQUEUE 341
  CEMT INQUIRE transaction 335
  CEMT PERFORM STATISTICS 425
  CEMT SET transaction 512
TEMPLATENAME
  CEMT INQUIRE
    DOCTEMPLATE 192
  CEMT INQUIRE URIMAP 391
TEMPLATETYPE
  CEMT INQUIRE
    DOCTEMPLATE 192
TEMPORARY
  CEMT INQUIRE URIMAP 390
  CEMT SET URIMAP 525
temporary storage browse transaction, CEBR 75
TEMPSTORAGE
  CEMT INQUIRE transaction 341
  CEMT SET transaction 514
TER
  CEOT 534
TERM
  CEMT INQUIRE DB2CONN 163
  CEMT INQUIRE DB2ENTRY 175
  CEMT INQUIRE TASK 319
  CEMT SET DB2CONN 440
  CEMT SET DB2ENTRY 450
TERMID
  CEDF 100
  CEMT INQUIRE BRFACILITY 141
  CEMT INQUIRE TDQUEUE 341
  CEMT INQUIRE UOW 370
  CMSG 578
  CSFE ZCQTRACE 616
TERMID GT 4 CHAR AT XXXXX
  CMSG 590
TERMID NOT +/- AT xxxxx
  CMSG 590
TERMID/OPID
  CMSG 578
TERMID*LDC
  CMSG 586
TERMINAL
  CEMT DISCARD 126
  CEMT INQUIRE NETNAME 276
  CEMT INQUIRE TERMINAL 348
  CEMT INQUIRE transaction 342
  CEMT PERFORM STATISTICS 425
  CEMT SET transaction 515
terminal and system test transaction, CSFE 615
terminal builder trace 616
terminal list table (TLT) 5
terminal operator
  duties 4
  master 6
  paging 619
  supervisory 5
  transactions 4
  write to console operator (CWTO) 629
terminals
  3270 displays 577
  3767 keyboard/printer 577
  autopaging 627
  CEMT INQUIRE requests 342
  CEMT SET requests 512, 515
  fixed-format 537
  identifying 5
  message-switching transaction 577
  paging 7, 619
  sequential 577
  status 533
  test 615
terminate a logical message 7, 625
TERMINATED BY OPERATOR
  CMSG 590
TERMLIST
  CMSG 578
TERMLIST*LDC
  CMSG 586
TERMSTATUS
  CEMT INQUIRE BRFACILITY 141
  CEMT INQUIRE NETNAME 276
  CEMT INQUIRE TERMINAL 348
testing a terminal 615
THREADCOUNT
  CEMT INQUIRE JVMSERVER 254
THREADERROR
  CEMT INQUIRE DB2CONN 163
THREADLIMIT
  CEMT INQUIRE DB2CONN 163
  CEMT INQUIRE DB2ENTRY 177
  CEMT INQUIRE JVMSERVER 254
  CEMT SET DB2ENTRY 452
THREADS
  CEMT INQUIRE DB2CONN 163
  CEMT INQUIRE DB2ENTRY 177
THREADSAFE
  CEMT INQUIRE PROGRAM 289
THREADWAIT
  CEMT INQUIRE DB2CONN 163
  CEMT INQUIRE DB2ENTRY 178
  CEMT SET DB2ENTRY 452
TIME
  CEMT INQUIRE DISPATCHER 184
  CEMT INQUIRE MONITOR 265
  CEMT INQUIRE SYSTEM 316
  CEMT SET DISPATCHER 456
  CEMT SET SYSTEM 507
  CMSG 578
TIME ALREADY PASSED
  CMSG 590
TIME INVALID AT XXXXX
  CMSG 590
TIME NOT 4 CHARS AT XXXXX
  CMSG 590
time-of-day resetting 418
TIMEOUTINT
  CEMT INQUIRE WEB 395
  CEMT SET WEB 528
TIMESTAMP
  CEMT INQUIRE DJAR 189
TITLE
  CEMT PERFORM DUMP 414
TLT (terminal list table) 5
TO
  CEMT INQUIRE TASK 321
TOO MANY TBL SFX AT XXXXX
  CMSG 590
TOTAL
  CEMT INQUIRE JVMPOOL 251
TOTALJVMS
  CEMT INQUIRE CLASSCACHE 146
TP
  CEMT INQUIRE TASK 321
TPNAME
  CEMT INQUIRE PARTNER 278
TPOOL
  CEMT INQUIRE DB2ENTRY 178
  CEMT SET DB2ENTRY 452
TRA
  CEOT 534
trace control transaction 553
trace program exit routine 618
trademarks 644
TRANClass
  CEKL INQUIRE TASK 106
TRANDUMP
  CEMT INQUIRE
    TRDUMPCODE 359
  CEMT SET TRDUMPCODE 522
TRANDUMPING
  CEMT INQUIRE
    TRDUMPCODE 359
TRANID
  CEDX 103
  CEMT INQUIRE TASK 322
  CEMT INQUIRE TDQUEUE 341
TRANISOLATE
  CEMT INQUIRE SYSTEM 317
TRANSACTION
  CEMT DISCARD 126
  CEMT INQUIRE BRFACILITY 142
  CEMT INQUIRE NETNAME 277
  CEMT INQUIRE TERMINAL 349
  CEMT INQUIRE transaction 349
  CEMT INQUIRE
    TRANSACTION 356
  CEMT INQUIRE URIMAP 391
  CEMT PERFORM STATISTICS 425
  CEMT SET transaction 519
transaction routing, CRTE 613
transaction, BTS browser 59
transaction, indoubt test tool 561
transaction, load module map 565
transaction, monitoring facility 111
transaction, trace control 553
transactions
  all (list) 633
  canceling 2
  CBAM 59
  CCRL 61
  CEBT 77
  CECI 89

transactions *(continued)*
CECS 89
CEDA 91
CEDB 91
CEDC 91
CEDF 99
CEKL 105
CEMT 113
CEOT 533
CESL 539
CESN 545
CEST 551
CMSG 573
CPIA 599
CREA 601
CREC 601
CRTE 613
CSFE 615
CSPG 619
CWTO 629
DSNC 631
identification codes 4
initiating from console 9
invocation of 1
security key 4
transactions with operator interface 633
transactions with operator interface
CADP 37
CBAM 59
CDBC 63
CDBI 67
CDBM 69
CEMN 111
CESF 537
CETR 553
CIDP 559
CLER 567
CMAC 569
TRANSID
CEKL INQUIRE TASK 106
CEMT INQUIRE DB2TRAN 180
CEMT INQUIRE ENQ 208
CEMT INQUIRE REQUESTMODEL 301
CEMT INQUIRE TSQUEUE 367
CEMT INQUIRE UOW 371
CEMT INQUIRE UOWENQ 380
CEMT SET DB2TRAN 453
TRANSID option
INQUIRE WORKREQUEST command 404
TRANSMODE
CEMT INQUIRE EPADAPTER 216
TRDUMP
CEMT PERFORM STATISTICS 425
TRDUMPCODE
CEMT INQUIRE transaction 356
CEMT INQUIRE TRDUMPCODE 359
CEMT SET transaction 520
TRIGGERLEVEL
CEMT INQUIRE TDQUEUE 341
TRIGMONTASKS
CEMT INQUIRE MQCONN 266
TRPROF
CEMT INQUIRE TRANSACTION 356
CRTE transaction 613
TSMAININUSE
CEMT INQUIRE TEMPSTORAGE 342
TSMAINLIMIT
CEMT INQUIRE TEMPSTORAGE 342
TSMODEL
CEMT DISCARD 126
CEMT INQUIRE transaction 359
CEMT INQUIRE TSMODEL 362
TSO console as a CICS terminal 12
TSPOOL
CEMT INQUIRE transaction 362
CEMT INQUIRE TSPOOL 363
TSQ
CEMT INQUIRE ENQ 208
CEMT INQUIRE UOWENQ 381
TSQNAME
CEMT INQUIRE transaction 364
TSQUEUE
CEMT INQUIRE ATOMSERVICE 131
CEMT INQUIRE transaction 364
CEMT INQUIRE TSQUEUE 367
CEMT PERFORM STATISTICS 425
CEMT SET transaction 522
CEMT SET TSQUEUE 523
TSQUEUELIMIT
CEMT INQUIRE MONITOR 265
CEMT SET MONITOR 487
TSYSTEM option
INQUIRE WORKREQUEST command 404
TSYSTYPE option
INQUIRE WORKREQUEST command 405
TTI
CEMT INQUIRE NETNAME 277
CEMT INQUIRE TERMINAL 349
CEMT SET NETNAME 492
CEMT SET TERMINAL 518
CEOT 534, 535
TTISTATUS
CEMT INQUIRE NETNAME 277
CEMT INQUIRE TERMINAL 349
TTTT
CSPG 626
TWAIT
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE DB2ENTRY 178
CEMT SET DB2ENTRY 452
TX
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE DB2ENTRY 175
CEMT SET DB2CONN 440
CEMT SET DB2ENTRY 450
TXID
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE DB2ENTRY 174
CEMT SET DB2CONN 440
CEMT SET DB2ENTRY 449
TYPE
CEMT INQUIRE DOCTEMPLATE 192
CEMT INQUIRE ENQ 208
CEMT INQUIRE JMODEL 246
CEMT INQUIRE JOURNALNAME 248
CEMT INQUIRE REQUESTMODEL 301
CEMT INQUIRE TDQUEUE 341
CEMT INQUIRE UOWENQ 380
CEMT INQUIRE UOWLINK 384

U

U
CEMT INQUIRE TASK 321
UDATAKEY
CEMT INQUIRE TRANSACTION 355
UDSASIZE
CEMT INQUIRE DSAS 195
UEXECKEY
CEMT INQUIRE PROGRAM 289
CEMT PERFORM JVMPOOL command 417
UKOPEN
CEMT INQUIRE TASK 321
UNATTEMPTED
CEMT INQUIRE CONNECTION 153
UNAUTH
CEMT INQUIRE CORBASERVER 163
UNAVAILABLE
CEMT INQUIRE DSNAME 199
CEMT INQUIRE RRMS 302
CEMT INQUIRE UOWLINK 384
CEMT INQUIRE VTAM 393
CEMT SET DSNAME 462
INQUIRE TCPIPSERVICE 331
UNBALANCD PARENS AT XXXXX
CMSG 590
UNCONNECTED
CEMT INQUIRE TSPOOL 363
CEMT INQUIRE UOWLINK 384
UNDEFINED
CEMT INQUIRE UOWDSNFAIL 374
UNDETERMINED
CEMT INQUIRE DSNAME 202
UNENABLED
CEMT INQUIRE FILE 226
unit of work resynchronization transaction (CPIA) 599
UNKNOWN
CEMT INQUIRE UOWLINK 384
UNQUIESCED
CEMT INQUIRE DSNAME 201
CEMT SET DSNAME 462
UNREGISTERED
CEMT INQUIRE VTAM 393
INQUIRE TCPIPSERVICE 331
UNSUP OPRT TRMNL XXXXXXXX
CMSG 590
UNSUP TERMINAL XXXXXXXX
CMSG 590
UOW
CEMT INQUIRE DB2CONN 163

UOW *(continued)*
CEMT INQUIRE DB2ENTRY 174
CEMT INQUIRE ENQ 208
CEMT INQUIRE TASK 322
CEMT INQUIRE UOW 371
CEMT INQUIRE UOWDSNFAIL 377
CEMT INQUIRE UOWENQ 381
CEMT INQUIRE UOWLINK 384
CEMT SET DB2CONN 440
CEMT SET transaction 523
CEMT transaction 368
UOWACTION
CEMT INQUIRE CONNECTION 157
CEMT INQUIRE DSNAME 202
UOWDSNFAIL
CEMT INQUIRE transaction 373
CEMT INQUIRE UOWDSNFAIL 377
UOWENQ
CEMT transaction 377
UOWLINK
CEMT INQUIRE transaction 381
CEMT INQUIRE UOWLINK 384
CEMT SET transaction 524
UOWSTATE
CEMT INQUIRE UOW 371
UPDATE
CEMT INQUIRE FILE 229
CEMT SET FILE 471
UPDATEMODEL
CEMT INQUIRE FILE 229
UPDATESTATUS
CEMT INQUIRE FILE 229
uppercase input to transactions 3
URID
CEMT INQUIRE EXCI 220
CEMT INQUIRE UOWLINK 384
URIMAP
CEMT DISCARD 126
CEMT INQUIRE ATOMSERVICE 131
CEMT INQUIRE transaction 385
CEMT INQUIRE WEBSERVICE 400
CEMT PERFORM STATISTICS 425
CEMT SET transaction 524
URM
CEMT SET TCPIPSERVICE 512
USAGE
CEMT INQUIRE URIMAP 391
USECOUNT
CEMT INQUIRE PROGRAM 289
CEMT INQUIRE STREAMNAME 307
USERAUTH option
CEMT INQUIRE IPCONN command 241
USERID 543, 548
CEMT INQUIRE BRFACILITY 142
CEMT INQUIRE DB2CONN 163
CEMT INQUIRE DB2ENTRY 175
CEMT INQUIRE NETNAME 277
CEMT INQUIRE TASK 322
CEMT INQUIRE TDQUEUE 341
CEMT INQUIRE TERMINAL 349
CEMT INQUIRE UOW 371
CEMT INQUIRE URIMAP 391
CEMT SET DB2CONN 440
CEMT SET DB2ENTRY 450
USERTABLE
CEMT INQUIRE FILE 229
CEMT SET FILE 471

V

VALID
CEMT INQUIRE DSNAME 203
VALIDATIONST
CEMT INQUIRE WEBSERVICE 401, 529
CEMT INQUIRE XMLTRANSFORM 408
CEMT SET XMLTRANSFORM 531
VALIDITY
CEMT INQUIRE DSNAME 203
value
CEKL SET TASK 109
CEMT INQUIRE BEAN 138
CEMT INQUIRE BRFACILITY 140
CEMT INQUIRE CONNECTION 149
CEMT INQUIRE CORBASERVER 158
CEMT INQUIRE DJAR 186
CEMT INQUIRE DSNAME 196
CEMT INQUIRE FILE 221
CEMT INQUIRE HOST 232
CEMT INQUIRE IPCONN 234
CEMT INQUIRE MODENAME 261
CEMT INQUIRE NETNAME 272
CEMT INQUIRE PARTNER 277
CEMT INQUIRE PROFILE 286
CEMT INQUIRE PROGRAM 289
CEMT INQUIRE REQUESTMODEL 297
CEMT INQUIRE STREAMNAME 306
CEMT INQUIRE SYDUMPCODE 308
CEMT INQUIRE TASK 317
CEMT INQUIRE TCLASS 322
CEMT INQUIRE TDQUEUE 335
CEMT INQUIRE TERMINAL 343
CEMT INQUIRE TRANSACTION 350
CEMT INQUIRE TRDUMPCODE 357
CEMT INQUIRE TSMODEL 360
CEMT INQUIRE TSPOOL 363
CEMT INQUIRE TSQUEUE 364
CEMT INQUIRE UOW 368
CEMT INQUIRE UOWLINK 381
CEMT INQUIRE URIMAP 385
CEMT INQUIRE WORKREQUEST 402
CEMT SET ATOMSERVICE 427
CEMT SET BRFACILITY 430
CEMT SET CONNECTION 433
CEMT SET CORBASERVER 439
CEMT SET DOCTEMPLATE 457
CEMT SET DSNAME 459
CEMT SET ENQMODEL 464
CEMT SET EPADAPTER 465
CEMT SET EVENTBINDING 465
CEMT SET FILE 467
CEMT SET HOST 472
CEMT SET IPCONN 474
value *(continued)*
CEMT SET JOURNALNAME 479
CEMT SET LINE 483
CEMT SET MODENAME 484
CEMT SET NETNAME 491
CEMT SET PIPELINE 493
CEMT SET PROGRAM 496
CEMT SET SYDUMPCODE 502
CEMT SET TASK 508
CEMT SET TCLASS 509
CEMT SET TERMINAL 516
CEMT SET TRANSACTION 519
CEMT SET TRDUMPCODE 521
CEMT SET TSQUEUE 522
CEMT SET UOW 523
CEMT SET UOWLINK 524
CEMT SET WEBSERVICE 529
CEMT SET WORKREQUEST 530
CEMT SET XMLTRANSFORM 531
VALUE
CEMT SET BUNDLE 431
value option
CEMT INQUIRE REQUESTMODEL 299
values
hexadecimal 119
VERSION
CEMT INQUIRE EVENTBINDING 218
view Java debugging profile screen
application debugging profile manager
3270 interface 53
View Java profile page
application debugging profile manager
web interface 33
view LE debugging profile screen
application debugging profile manager
3270 interface 52
View LE profile page
application debugging profile manager
web interface 32
viewing debugging profiles
with the 3270 interface 42
with the web interface 23
VSAM
CEMT INQUIRE DSNAME 197
CEMT INQUIRE FILE 223
VTAM
CEMT INQUIRE VTAM 394
CEMT PERFORM STATISTICS 425
VTAM ACB (access control block) 525

W

WAIT
CEMT INQUIRE TASK 320
CEMT INQUIRE TDQUEUE 339
CEMT INQUIRE TRANSACTION 353
WAITCAUSE
CEMT INQUIRE UOW 371
WAITCOMMIT
CEMT INQUIRE UOW 372

WAITER
CEMT INQUIRE ENQ 207
CEMT INQUIRE UOWENQ 380
WAITFORGET
CEMT INQUIRE UOW 372
WAITING
CEMT INQUIRE ENQMODEL 212
CEMT INQUIRE UOW 372
WAITRMI
CEMT INQUIRE UOW 372
WAITSTATE
CEMT INQUIRE UOW 372
web interface
application debugging profile manager 15
web support 528
CEMT INQUIRE transaction 394
CEMT SET transaction 528
web support inquiry 394
WEBSERVICE
CEMT DISCARD 126
CEMT INQUIRE URIMAP 391
CEMT PERFORM STATISTICS 425
CEMT SET WEBSERVICE 528
work requests
CEMT INQUIRE requests 401
CEMT SET requests 529
WORKREQUEST
CEMT SET transaction 529
WORKREQUEST command
CEMT INQUIRE transaction 401
WORKREQUEST option
INQUIRE WORKREQUEST command 405
WORKTYPE option
INQUIRE WORKREQUEST command 405
write to console operator (CWTO) 629
WSBIND
CEMT INQUIRE WEBSERVICE 401
WSDIR
CEMT INQUIRE PIPELINE 282
CEMT INQUIRE WEBSERVICE 399
WSDLFILE
CEMT INQUIRE WEBSERVICE 401

X

XCF
CEMT INQUIRE CONNECTION 151
XCFGROUP
CEMT INQUIRE IRC 243
XLNSTATUS
CEMT INQUIRE CONNECTION 157
XLT
CEMT PERFORM SHUTDOWN 420
XM
CEMT INQUIRE CONNECTION 151
XMLSCHEMA
CEMT INQUIRE XMLTRANSFORM 408
XMLTRANSFORM
CEMT INQUIRE ATOMSERVICE 131
CEMT INQUIRE transaction 405
CEMT PERFORM STATISTICS 425
CEMT SET WEBSERVICE 531
XNOTDONE
CEMT INQUIRE CONNECTION 158
XOK
CEMT INQUIRE CONNECTION 158
XOPDIRECTST
CEMT INQUIRE PIPELINE 282
CEMT INQUIRE WEBSERVICE 401
XOPSUPPORTST
CEMT INQUIRE PIPELINE 282
CEMT INQUIRE WEBSERVICE 401
XRF and the CEBT transaction 77
XSDBIND
CEMT INQUIRE XMLTRANSFORM 408

Y

YES
CMSG 578
YY.DDD
CMSG 578
YYYY
CSPG 623
YYYY.DDD
CMSG 578

Z

z/OS Communications Server
CEMT INQUIRE CONNECTION 151
CEMT INQUIRE transaction 392
CEMT SET transaction 525
z/OS Communications Server ACB (access control block) 392
z/OS Communications Server terminals, acquiring 491, 516
z/OS UNIXFILE
CEMT INQUIRE DJAR 188
ZERO LENGTH MSG AT ''
CMSG 590

Readers' Comments — We'd Like to Hear from You

CICS Transaction Server for z/OS
Version 4 Release 2
Supplied Transactions

Publication No. SC34-7184-00

We appreciate your comments about this publication. Please comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. The comments you send should pertain to only the information in this manual or product and the way in which the information is presented.

For technical questions and information about products and prices, please contact your IBM branch office, your IBM business partner, or your authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you. IBM or any other organizations will only use the personal information that you supply to contact you about the issues that you state on this form.

Comments:

Thank you for your support.

Submit your comments using one of these channels:
- Send your comments to the address on the reverse side of this form.
- Send a fax to the following number: +44 1962 816151
- Send your comments via email to: idrcf@uk.ibm.com

If you would like a response from IBM, please fill in the following information:

Name	Address
Company or Organization	
Phone No.	Email address

Readers' Comments — We'd Like to Hear from You
SC34-7184-00

Cut or Fold
Along Line

Fold and Tape | **Please do not staple** | Fold and Tape

PLACE
POSTAGE
STAMP
HERE

IBM United Kingdom Limited
User Technologies Department (MP095)
Hursley Park
Winchester
Hampshire
United Kingdom
SO21 2JN

Fold and Tape | **Please do not staple** | Fold and Tape

SC34-7184-00

Cut or Fold
Along Line

SC34-7184-00