IMS Version 9

Command Reference
Build Date: 10/16/03

IMS Version 9

Command Reference
Build Date: 10/16/03

Note
Before using this information and the product it supports, be sure to read the general information under “Notices” on page xxi

Quality Partnership Program (QPP) Edition (January 2004)

This QPP edition applies to Version 9 of IMS (product number 5655-J38) and to all subsequent releases and modifications until otherwise indicated in new editions.

© Copyright International Business Machines Corporation 1974, 2003. All rights reserved.
US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Figures xv

Tables xvii

Notices xxi
Programming Interface Information xxiii
Trademarks. xxiii
Product Names xxiv

Preface xxv
Summary of Contents xxv
Prerequisite Knowledge xxvi
Syntax Diagrams xxvi
How to Send Your Comments xxviii
Change Indicators xxviii

Summary of Changes xxix
Changes to This Book for IMS Version 9 xxix
Library Changes for IMS Version 9 xxix

Part 1. Introduction 1

Chapter 1. IMS Command Language 3
Introduction 3
How to Enter an IMS Command 3
Commands from the OM API. 12
Command Responses to OM. 13
Command Responses 14
Responses to LU 6.2 Devices 14
Sending Messages to the z/OS System Console 15
Sending Messages to the IMS Master Terminal 15
IMS Terminal Command Examples 15
TSO SPOC Command Examples 16
OM API Command Examples 16
IMS Classic Command Format 16
Command Processing in an IMSplex 21
IMSplex Command Format 22
Command Characteristics 24
Terminal Security Defaults. 24
Commands Recovered During Emergency Restart. 26
Commands Logged to the Secondary Master Terminal 27
Commands Supported from an AO Application 27
Command Security When Using OM 33
Commands Mirrored on an XRF Alternate 33
Commands Supported on the XRF Alternate 34
Commands and Keywords Supported on an RSR Tracking Subsystem 36
BTAM Terminals in an RSR Environment 38
Commands Supported from LU 6.2 Devices and OTMA 38
Terminal Security Defaults for LU 6.2 Devices and OTMA 39
Commands Supported by the OM API 39

Chapter 2. Keyword Table and Definitions 43

© Copyright IBM Corp. 1974, 2003

Part 2. IMS Commands 79

Chapter 3. /ACTIVATE 95
Format 95
Environments and Keywords 95
Usage 95
Examples 95

Chapter 4. /ALLOCATE 97
Format 97
Environments and Keywords 97
Usage 97

Chapter 5. /ASSIGN 99
Format 99
Environments and Keywords 100
Usage. 101
Examples 110

Chapter 6. /BROADCAST 115
Format 115
Environments and Keywords 115
Usage. 116
Examples 117

Chapter 7. /CANCEL 121
Format 121
Environments 121
Usage. 121
Example for /CANCEL Command 121

Chapter 8. /CHANGE 123
Format 123
Environments and Keywords 124
Usage. 125
Examples 133

Chapter 9. /CHECKPOINT 139
Format 139
Environments and Keywords 139
Usage. 140
Examples 143

Chapter 10. /CLSDST 145
Format 145
Environments and Keywords 145
Usage. 145
Example for /CLSDST Command. 147

Chapter 11. /COMPT 149
Format 149
Environments and Keywords 149
Usage. 150
Examples 150

Chapter 12. /CQCHKPT 153

Format 153
Environments and Keywords 153
Usage. 153
Examples 154

Chapter 13. /CQQUERY 155
Format 155
Environments and Keywords 155
Usage. 155
Examples 156

Chapter 14. /CQSET 157
Format 157
Environments and Keywords 157
Usage. 157
Example for /CQSET Command 158

Chapter 15. /DBDUMP 159
Format 159
Environments and Keywords 159
Usage. 159
Examples 161

Chapter 16. /DBRECOVERY 163
Format 163
Environments and Keywords 163
Usage. 163
Examples 167

Chapter 17. /DELETE 171
Format 171
Environments and Keywords 171
Usage. 172
Examples 172

Chapter 18. DELETE 175
Format 175
Environments and Keywords 175
Usage. 175
Examples 177

Chapter 19. /DEQUEUE 183
Format 183
Environments and Keywords 183
Usage. 184
Examples 188

Chapter 20. /DIAGNOSE 191
Format 191
Environments 191
Usage. 191
Example for /DIAGNOSE Command 194

Chapter 21. /DISPLAY 195
Format 195
Environments and Keywords 209

Usage . . . 211
/DISPLAY ACTIVE . . . 211
/DISPLAY AFFINITY . . . 216
/DISPLAY AOITOKEN . . . 216
/DISPLAY APPC . . . 216
/DISPLAY AREA . . . 218
/DISPLAY ASSIGNMENT . . . 220
/DISPLAY CCTL . . . 221
/DISPLAY CONVERSATION . . . 222
/DISPLAY CPLOG . . . 223
/DISPLAY CQS . . . 223
/DISPLAY DATABASE . . . 223
/DISPLAY DATABASE with BKERR . . . 225
/DISPLAY DBD . . . 226
/DISPLAY DESCRIPTOR . . . 226
/DISPLAY FDR . . . 226
/DISPLAY FPVIRTUAL . . . 227
/DISPLAY HSB . . . 228
/DISPLAY HSSP . . . 229
/DISPLAY LINE . . . 230
/DISPLAY LINK . . . 230
/DISPLAY LTERM . . . 231
/DISPLAY LUNAME . . . 232
/DISPLAY MASTER . . . 234
/DISPLAY MODIFY . . . 234
/DISPLAY MSNAME . . . 239
/DISPLAY NODE . . . 239
/DISPLAY OASN SUBSYS . . . 242
/DISPLAY OLDS . . . 242
/DISPLAY OTMA . . . 244
/DISPLAY OVERFLOWQ . . . 245
/DISPLAY POOL . . . 246
/DISPLAY PROGRAM . . . 254
/DISPLAY PSB . . . 255
/DISPLAY PTERM . . . 255
/DISPLAY Q . . . 256
/DISPLAY QCNT . . . 256
/DISPLAY RECOVERY . . . 258
/DISPLAY RTCODE . . . 261
/DISPLAY SHUTDOWN STATUS . . . 261
/DISPLAY STATUS (With No Keywords) . . . 263
/DISPLAY STRUCTURE . . . 265
/DISPLAY SUBSYS . . . 265
/DISPLAY SYSID TRANSACTION . . . 267
/DISPLAY TIMEOVER . . . 267
/DISPLAY TMEMBER . . . 267
/DISPLAY TRACE . . . 269
/DISPLAY TRACKING STATUS . . . 270
/DISPLAY TRANSACTION . . . 278
/DISPLAY UOR . . . 279
/DISPLAY USER . . . 280
Examples . . . 282

Chapter 22. /END . . . 351
Format . . . 351
Environments and Keywords . . . 351

Usage. 351
Example for /END Command 352

Chapter 23. /ERESTART 353
Format 353
Environments and Keywords 354
Usage. 355
Examples 361

Chapter 24. /EXCLUSIVE 365
Format 365
Environments and Keywords 365
Usage. 365
Examples 366

Chapter 25. /EXIT 367
Format 367
Environments and Keywords 367
Usage. 367
Examples 368

Chapter 26. /FORMAT 371
Format 371
Environments and Keywords 371
Usage. 371
Example for /FORMAT Command 372

Chapter 27. /HOLD. 373
Format 373
Environments 373
Usage. 373
Example for /HOLD Command 373

Chapter 28. /IAM 375
Format 375
Environments and Keywords 375
Usage. 375
Examples 376

Chapter 29. /IDLE 377
Format 377
Environments and Keywords 377
Usage. 377
Examples 379

Chapter 30. /LOCK. 381
Format 381
Environments and Keywords 381
Usage. 381
Examples 383

Chapter 31. INITIATE 385
Format 385
Environments and Keywords 385
INITIATE OLC. 386

Chapter 32. /LOG 413
Format 413
Environments 413
Usage. 413
Example for /LOG Command 413

Chapter 33. /LOOPTEST 415
Format 415
Environments and Keywords 415
Usage. 415
Example for /LOOPTEST Command 415

Chapter 34. /MODIFY 417
Format 417
Environments and Keywords 417
Usage. 417
Examples 421

Chapter 35. /MONITOR 425
Format 425
Environments and Keywords 425
Usage. 425
Example for /MONITOR Command 425

Chapter 36. /MSASSIGN 427
Format 427
Environments and Keywords 427
Usage. 427
Examples 428

Chapter 37. /MSVERIFY 431
Format 431
Environments and Keywords 431
Usage. 431
Examples 431

Chapter 38. /NRESTART 435
Format 435
Environments and Keywords 437
Usage. 437
Examples 442

Chapter 39. /OPNDST 447
Format 447
Environments and Keywords 447
Usage. 448
Examples 450

Chapter 40. /PSTOP 453
Format 453
Environments and Keywords 453
Usage. 454
Examples 456

Chapter 41. /PURGE 461
Format 461

Environments and Keywords 461
Usage. 462
Examples 463

Chapter 42. QUERY 465
Format 465
Environments and Keywords 468
QUERY AREA 470
QUERY DB. 478
QUERY IMSPLEX 487
QUERY LE 495
QUERY MEMBER 501
QUERY OLC 509
QUERY STRUCTURE. 516
QUERY TRAN 520

Chapter 43. /QUIESCE 537
Format 537
Environments and Keywords 537
Usage. 537
Example for /QUIESCE Command 538

Chapter 44. /RCLSDST 539
Format 539
Environments 539
Usage. 539
Example for /RCLSDST Command 540

Chapter 45. /RCOMPT 541
Format 541
Environments and Keywords 541
Usage. 541
Example for /RCOMPT Command 542

Chapter 46. /RDISPLAY 543
Format 543
Environments 543
Usage. 543
Examples 543

Chapter 47. /RECOVER 545
Format 545
Environments and Keywords 546
Usage. 547
Examples 556

Chapter 48. /RELEASE 561
Format 561
Environments and Keywords 561
Usage. 561
Example for /RELEASE Command 561

Chapter 49. /RESET 563
Format 563
Environments 563
Usage. 563

Example for /RESET Command 563

Chapter 50. /RMxxxxxx 565
Format 565
Environments and Keywords 565
Usage. 565
Examples 567

Chapter 51. /RSTART 573
Format 573
Environments and Keywords 573
Usage. 574
Examples 576

Chapter 52. /RTAKEOVER 579
Format 579
Environments and Keywords 579
Usage. 579
Examples 580

Chapter 53. /SECURE 583
Format 583
Environments and Keywords 583
Usage. 583
Examples 584

Chapter 54. /SET 587
Format 587
Environments and Keywords 587
Usage. 587
Examples 588

Chapter 55. /SIGN 591
Format 591
Environments 591
Usage. 591
Examples 594

Chapter 56. /SMCOPY 597
Format 597
Environments and Keywords 597
Usage. 597
Example for /SMCOPY Command 598

Chapter 57. /SSR 599
Format 599
Environments 599
Usage. 599
Example for /SSR Command 599

Chapter 58. /START 601
Format 601
Environments and Keywords 604
Usage. 605
Examples 620

Chapter 59. /STOP 629
Format 629
Usage. 632
Examples 644

Chapter 60. /SWITCH 657
Format 657
Environments and Keywords 657
Usage. 657
Examples 658

Chapter 61. TERMINATE 661
Format 661
Environments and Keywords 661
TERMINATE OLC 661

Chapter 62. /TEST 673
Format 673
Environments and Keywords 673
Usage. 673
Examples 675

Chapter 63. /TRACE 677
Format 677
Environments and Keywords 680
Usage. 681
Examples 693

Chapter 64. /UNLOCK 699
Format 699
Environments and Keywords 699
Usage. 699
Examples 701

Chapter 65. UPDATE 705
Format 705
Environments and Keywords 707
2 UPDATE AREA 709
| UPDATE DATAGRP 716
| UPDATE DB 723
UPDATE LE 738
UPDATE TRAN 742

Chapter 66. /VUNLOAD 751
Format 751
Environments and Keywords 751
Usage. 751
| Example for /VUNLOAD Command 752

Part 3. z/OS (MVS) Commands Used for IMS 753

Chapter 67. Introduction 755

Chapter 68. START FDBRPROC 757
Format 757
Usage. 757

Chapter 69. MODIFY IMS . . . 759
F jobname,DUMP . . . 759
F jobname,DUMPxxxx . . . 759
F jobname,FORCExxxx . . . 760
F jobname,RECONNECT . . . 761
F jobname,RECONNSTR . . . 762
F jobname,STOP . . . 763
F jobname,STOPxxxx . . . 763

Chapter 70. START IRLMPROC . . . 765
Format . . . 765
Usage. . . . 765

Chapter 71. MODIFY FDBRPROC . . . 769
F fdbrproc,DUMP . . . 769
F fdbrproc,RECOVER . . . 769
F fdbrproc,STATUS . . . 769
F fdbrproc,STOP. . . . 770
F fdbrproc,TERM . . . 771

Chapter 72. MODIFY IRLMPROC . . . 773
F irlmproc,ABEND . . . 773
F irlmproc,DIAG,HANG . . . 774
F irlmproc,PURGE,imsname . . . 775
F irlmproc,SET . . . 775
F irlmproc,STATUS . . . 779

Chapter 73. STOP CQSJOB . . . 783
Format . . . 783
Usage. . . . 783

Chapter 74. STOP IRLMPROC . . . 785
Format . . . 785
Usage. . . . 785
Example . . . 785

Chapter 75. TRACE CT . . . 787
Format . . . 787
Usage. . . . 787
Examples . . . 789

Chapter 76. CANCEL/FORCE ODBA . . . 791
Format . . . 791
Usage. . . . 791
Format . . . 791
Usage. . . . 791

Chapter 77. STOP CSL Address Spaces . . . 793
Format . . . 793
Usage. . . . 793

Part 4. IMS Transport Manager Subsystem Commands . . . 795

Chapter 78. Introduction . . . 797

Chapter 79. DEFINE . . . 799

Format 799
Usage. 799

Chapter 80. DISPLAY 801
Format 801
Usage. 801
Examples 805

Chapter 81. SET 807
Format 807
Usage. 807

Chapter 82. START 809
Format 809
Usage. 809

Chapter 83. STOP 811
Format 811
Usage. 811

Part 5. Base Primitive Environment Commands 813

Chapter 84. BPE Commands 815
BPE Command Syntax and Invocation. 815
BPE TRACETABLE Commands 817
BPE DISPLAY VERSION Command 824

Part 6. Appendixes 837

Appendix A. DBCTL Commands 839

Appendix B. DCCTL Commands 841

Appendix C. List of Reserved Words 845

Appendix D. Shared Secondary Index Database Commands 849

Appendix E. Commands with the NODE USER Keyword Combination 851

Appendix F. Commands That Are Valid in ETO 853

Appendix G. Status and Attributes for the /DISPLAY Command 855

Appendix H. High Availability Large Database Commands 867

Appendix I. IMS Commands, RACF Access Authorities and Resource Names Table 869

Bibliography 877
IMS Version 9 Library 877

Figures

1. Master Terminal Format 4
2. Master Terminal Format in XRF Environment 5
3. TSO SPOC Screen Format 12
4. IMS Formatted Master for the Alternate System 363
5. IMS Formatted Master for the Active System 363
6. Alternate System at Start of Takeover 659
7. Newly Created Active System after Takeover 660
8. I/O Toleration Phase of Processing 702
9. /UNLOCK SYSTEM Command 703
10. EEQE Deleted as Part of /UNLOCK SYSTEM Processing 703

© Copyright IBM Corp. 1974, 2003

Tables

1 1. Command and Keyword Changes for IMS Version 9 xxix
2. Commands and Keywords that Support Generic Parameters. 19
3. Terminal Security Defaults for IMS Commands 24
4. Commands Recovered During Emergency Restart 26
5. Commands Logged to the Secondary Master Terminal 27
6. Commands Supported from an AO Application Using CMD 28
7. Commands Supported from an AO Application Using ICMD 30
8. List of IMS Commands Mirrored on an XRF Alternate 33
9. List of Commands Supported on the XRF Alternate 34
10. Commands and Keywords Allowable on an IMS DB/DC RSR Tracking Subsystem 36
11. Commands Supported from LU 6.2 Devices and OTMA 38
12. Terminal Security Defaults for IMS Commands from LU 6.2 Devices and OTMA 39
13. Commands Supported by the OM API 40
2 14. Keywords, Synonyms, and Their Environments. 43
15. Commands That Support the USER Keyword 76
16. Valid Environments for the /ACTIVATE Command and Keywords 95
17. Valid Environments for the /ALLOCATE Command and Keywords 97
18. Valid Environments for the /ASSIGN Command and Keywords 100
19. Relationships Between COMPONENT/ICOMPONENT Values and Terminals 104
20. Valid Environments for the /BROADCAST Command and Keywords 115
21. Valid Environments for the /CANCEL Command 121
22. Valid Environments for the /CHANGE Command and Keywords 124
23. Changing the TIMEOUT Value 131
24. Changing the INTERVAL Value 131
25. Valid Environments for the /CHECKPOINT Command and Keywords 139
26. IMS Shutdown Options and Their Effect on IMS Resources. 141
27. Valid Environments for the /CLSDST Command and Keywords 145
28. Valid Environments for the /COMPT Command and Keywords. 149
29. Valid Environments for the /CQCHKPT Command and Keywords 153
30. Valid Environments for the /CQQUERY Command and Keywords 155
31. Valid Environments for the /CQSET Command and Keywords 157
32. Valid Environments for the /DBDUMP Command and Keywords 159
33. Return and reason code for the GLOBAL keyword issued from the OM API 161
34. Valid Environments for the /DBRECOVERY Command and Keywords 163
35. Return and reason code for GLOBAL keyword issued from the OM API 166
36. Valid Environments for the /DELETE Command and Keywords 171
37. Valid Environments for the DELETE Command and Keywords 175
38. Output Fields for the DELETE LE Command 176
39. Return and Reason Codes for the DELETE LE Command 177
40. Completion Code for the DELETE LE Command 177
41. Parameter Override Table Entries for DELETE LE Example. 177
42. Example Filters and Resulting Actions for DELETE LE Command 178
43. Valid Environments for the /DEQUEUE Command and Keywords 183
44. Valid Environments for the /DIAGNOSE Command and Keywords 191
45. Valid Environments for the /DISPLAY Command and Keywords 209
46. /DISPLAY AREA Command Attributes. 220
47. /DISPLAY DATABASE Command Attributes 225
48. /DISPLAY LINE Command Attributes 230
49. Work in Progress For Resources that Prevent a Sucessful /MODIFY COMMIT 235
50. Parameters Supported in a DBCTL Environment. 235
51. Parameters Supported in a DCCTL Environment. 236
52. Storage Pools and Their Environments 248
53. /DISPLAY PTERM Command Attributes 255

© Copyright IBM Corp. 1974, 2003

54. /DISPLAY STATUS Resource States Displayed (No Keywords) 263
55. Display Exit Entry Points for Tracing 269
56. Valid Environments for the /END Command and Keywords 351
57. Valid Environments for the /ERESTART Command and Keywords 354
58. Security Keywords and Their Startup Parameter Equivalents 357
59. /ERESTART FORMAT Command Parameter Environments 358
60. Valid Environments for the /EXCLUSIVE Command and Keywords 365
61. Valid Environments for the /EXIT Command and Keywords 367
62. Valid Environments for the /FORMAT Command and Keyword 371
63. Valid Environments for the /HOLD Command 373
64. Valid Environments for the /IAM Command and Keywords 375
65. Valid Environments for the /IDLE Command and Keywords 377
66. Valid Environments for the /LOCK Command and Keywords 381
67. Valid Environments for the INITIATE OLC Command and Keywords 385
68. INITIATE OLC Output Field Descriptions. 393
69. Return and Reason Codes for INITIATE OLC 394
70. Completion Codes for the INITIATE OLC Command 398
71. Valid Environments for the /LOG Command 413
72. Valid Environments for the /LOOPTEST Command and Keywords 415
73. Valid Environments for the /MODIFY Command and Keywords 417
74. Resources and Parameters Available to the /MODIFY Command 418
75. /MODIFY PREPARE Parameters Environments 420
76. Valid Environments for the /MONITOR Command and Keywords 425
77. Valid Environments for the /MSASSIGN Command and Keywords 427
78. Valid Environments for the /MSVERIFY Command and Keywords 431
79. Valid Environments for the /NRESTART Command and Keywords 437
80. Security Keywords and Their Startup Parameter Equivalents 438
81. /NRESTART FORMAT Command Parameter Environments 440
82. Valid Environments for the /OPNDST Command and Keywords 447
83. Valid Environments for the /PSTOP Command and Keywords 453
84. Valid Environments for the /PURGE Command and Keywords. 461
I 85. Valid Environments for QUERY AREA Command and Keywords 468
I 86. Valid Environments for QUERY DB Command and Keywords 468
87. Valid Environments for QUERY IMSPLEX Command and Keywords 469
88. Valid Environments for the QUERY LE Command and Keywords. 469
89. Valid Environments for the QUERY MEMBER Command and Keywords 469
90. Valid Environments for the QUERY OLC Command and Keywords 469
91. Valid Environments for the QUERY STRUCTURE Command and Keywords 469
92. Valid Environments for the QUERY TRAN Command and Keywords 470
3 93. Output Fields for QUERY AREA Command. 472
I 94. Status Conditions for QUERY AREA 473
I 95. ADS Status Conditions for QUERY AREA 474
I 96. Return and Reason Code for the QUERY AREA Command. 474
I 97. Completion Codes for the QUERY AREA Command 475
I 98. QUERY AREA Compared to Similar IMS Commands 475
3 99. Output Fields for QUERY DB Command. 479
I 100. Database Status Conditions for the QUERY DB Command 481
I 101. Return and Reason Codes for the QUERY DB Command 481
I 102. Completion Codes for the QUERY DB Command 482
I 103. QUERY DB Command Compared to Similar Commands. 482
104. Output Fields for QUERY IMSPLEX Command 489
105. QUERY IMSPLEX Status Table 489
106. QUERY IMSPLEX Member Types 490
107. QUERY IMSPLEX Member Subtypes 490
108. Return and Reason Codes for the QUERY IMSPLEX Command 492
109. Completion Codes for the QUERY IMSPLEX Command 492

110. Output Fields for QUERY LE Command 497
111. Return and Reason Codes for the QUERY LE command 497
112. Completion code for the QUERY LE command 497
113. Parameter Override Table Entries for Example 1 498
114. Output Fields for QUERY MEMBER Command 502
115. Attributes for QUERY MEMBER Command 503
116. Status for QUERY MEMBER Command 503
117. Return and Reason Codes for QUERY MEMBER Command 506
118. Completion Codes for QUERY MEMBER 506
119. Output Fields for the QUERY OLC Command 510
120. Return and Reason Codes for the QUERY OLC Command 513
121. Completion Codes for the QUERY OLC Command 513
122. Output Fields for the QUERY STRUCTURE Command 517
123. Return and Reason Codes for the QUERY STRUCTURE Command 519
124. Completion Codes for the QUERY STRUCTURE Command 519
125. QUERY TRAN Compared to Other Similar Commands 524
126. Output Fields for the QUERY TRAN Command 524
127. Return and Reason Codes for the QUERY TRAN Command 528
128. Completion Codes for the QUERY TRAN Command 529
129. Valid Environments for the /QUIESCE Command and Keywords 537
130. Valid Environments for the /RCLSDST Command 539
131. Valid Environments for the /RCOMPT Command and Keywords 541
132. Valid Environments for the /RDISPLAY Command and Keyword 543
133. Valid Environments for the /RECOVER Command and Keywords 546
134. Valid Environments for the /RELEASE Command and Keyword 561
135. Valid Environments for the /RESET Command 563
136. Valid Environments for the /RMxxxxxx Command and Keyword 565
137. Functions of the DBRC Commands Supported Online 566
138. DBRC Modifiers for the /RMxxxxxx Commands 566
139. Valid Environments for the /RSTART Command and Keywords 573
140. Valid Environments for the /RTAKEOVER Command and Keywords 579
141. Valid Environments for the /SECURE Command and Keywords 583
142. Valid Environments for the /SET Command and Keywords 587
143. Valid Environments for the /SIGN Command 591
144. Valid Environments for the /SMCOPY Command and Keywords 597
145. Commands that Are Logged to the Secondary Master Terminal 597
146. Valid Environments for the /SSR Command 599
147. Valid Environments for the /START Command and Keywords 604
148. Return and reason code for GLOBAL keyword issued from the OM API 607
149. Return and reason code for GLOBAL keyword issued from the OM API 610
150. Valid Environments for the /STOP Command and Keywords 631
151. Return and reason code for GLOBAL keyword issued from the OM API 634
152. Valid Environments for the /SWITCH Command and Keywords 657
153. Valid Environments for the TERMINATE OLC Command 661
154. Output Fields for the TERMINATE Command 662
155. Return and Reason Codes for the TERMINATE Command 663
156. Completion Codes for the TERMINATE Command 667
157. Valid Environments for the /TEST Command and Keywords 673
158. Valid Environments for the /TRACE Command and Keywords 680
159. MONITOR Keyword Parameter Environments 685
160. /TRACE Command Formats 686
161. Trace Tables and Environments in Which They Are Valid 689
162. UNITYPE Parameters and Terminal Types 693
163. Valid Environments for the /UNLOCK Command and Keywords 699
| 164. Valid Environments for the UPDATE AREA Command and Keywords 707
| 165. Valid Environments for the UPDATE DATAGRP Command and Keywords 708

166. Valid Environments for the UPDATE DB Command and Keywords . . . 708
167. Valid Environments for the UPDATE LE Command and Keywords . . . 708
168. Valid Environments for the UPDATE TRAN Command and Keywords . . . 708
169. Output Fields for the UPDATE AREA Command . . . 711
170. Return and Reason Codes for the UPDATE AREA Command . . . 712
171. Completion Codes for the UPDATE AREA Command . . . 713
172. UPDATE AREA Command Compared to Other Similar IMS Commands . . . 714
173. Output Fields for UPDATE DATAGRP Command . . . 719
174. Return and Reason Codes for the UPDATE DATAGRP Command . . . 719
175. Completion Codes for the UPDATE DATAGRP Command . . . 720
176. UPDATE DATAGRP Command Compared to Other IMS Commands . . . 721
177. UPDATE DB Output Fields . . . 729
178. Return and Reason Codes for UPDATE DB . . . 729
179. Completion Codes for UPDATE DB . . . 731
180. UPDATE DB Command Compared to Other IMS Commands . . . 734
181. Output Fields for the UPDATE LE Command . . . 740
182. Return and Reason Codes for the UPDATE LE Command . . . 740
183. Completion Code for the UPDATE LE Command . . . 740
184. Parameter Override Table Entries for UPDATE LE Example 1 . . . 741
185. Output Fields for the UPDATE TRAN Command . . . 745
186. Return and Reason Codes for the UPDATE TRAN Command . . . 745
187. Completion Code for the UPDATE TRAN Command . . . 747
188. UPDATE TRAN Compared to Other Commands . . . 747
189. Valid Environments for the /VUNLOAD Command and Keyword . . . 751
190. Valid Values for OWNER Parameter . . . 817
191. Commands and Keywords Valid in DBCTL . . . 839
192. Commands and Keywords Valid in DCCTL . . . 841
193. Words Used and Reserved Only for IMS Commands . . . 845
194. Commands Valid in ETO . . . 853
195. Attributes and Status in the /DISPLAY Command . . . 855
196. Resource Names and RACF Authority for IMS Commands . . . 869

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 to the United Kingdom or any other country where such provisions are inconsistent with local law:
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs

© Copyright IBM Corp. 1974, 2003

and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
J46A/G4
555 Bailey Avenue
San Jose, CA 95141-1003
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Programming Interface Information

This book is intended to help terminal operators use the IMS Version 9 commands and the z/OS commands used for the Internal Resource Lock Manager. This book primarily documents General-use Programming Interface and Associated Guidance Information provided by IMS Version 9.

General-use programming interfaces allow the customer to write programs that obtain the services of IMS Version 9.

However, this book also documents Product-sensitive Programming Interface and Associated Guidance Information provided by IMS Version 9.

Product-sensitive programming interfaces allow the customer installation to perform tasks such as diagnosing, modifying, monitoring, repairing, tailoring, or tuning of IMS Version 9. Use of such interfaces creates dependencies on the detailed design or implementation of the IBM software product. Product-sensitive programming interfaces should be used only for these specialized purposes. Because of their dependencies on detailed design and implementation, it is to be expected that programs written to such interfaces may need to be changed in order to run with new product releases or versions, or as a result of service.

Product-sensitive programming interface

Product-sensitive Programming Interface and Associated Guidance Information is identified where it occurs, either by an introductory statement to a chapter or section or by the markup that surrounds this paragraph.

End of Product-sensitive programming interface

Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

ACF/VTAM
BookManager
CICS
Hiperspace
IBM
IMS
IMS/ESA
MVS/ESA
RACF
VTAM
z/OS

Java and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc., in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, and service names may be trademarks or service marks of others.

Product Names

In this book, the licensed program “DB2 Universal Database for z/OS” is referred to as “DB2.”

Preface

This book provides terminal operators with the information needed to use:

- IMS Commands
- z/OS Commands used for IMS
- Transport Manager Subsystem (TMS) commands
- Base Product Environment (BPE) commands

Because each user's IMS system is tailored to a specific operating configuration, this book is not intended to serve each installation's operational needs. It is recommended that each installation prepare an operator's guide to meet the needs of its IMS terminal operators. Use the contents of this manual to create or supplement the guide your installation prepares.

This book is available on the IMS Version 9 Licensed Product Kit (LK3T-7092). You can also get the most current versions of the PDF and BookManager formats by going to the IMS Web site at www.ibm.com/software/data/ims and linking to the Library page.

Summary of Contents

This book contains:

- Part 1, "Introduction," on page 1 introduces the IMS command language, including the format, keywords, parameters, syntax notation, and examples for the IMS commands.
- Part 2, "IMS Commands," on page 79 describes the IMS commands, in alphabetical order, and includes syntax diagrams.
- Part 3, "z/OS (MVS) Commands Used for IMS," on page 753 describes the z/OS commands used with the Internal Resource Lock Manager (IRLM) and includes the syntax diagrams.
- Part 4, "IMS Transport Manager Subsystem Commands," on page 795 describes the IMS Transport Manager Subsystem commands and includes the syntax diagrams.
- Part 5, "Base Primitive Environment Commands," on page 813 describes the general Base Product Environment (BPE) commands.
- A set of appendixes:
  - Appendix A, "DBCTL Commands," on page 839
  - Appendix B, "DCCTL Commands," on page 841
  - Appendix C, "List of Reserved Words," on page 845
  - Appendix D, "Shared Secondary Index Database Commands," on page 849
  - Appendix E, "Commands with the NODE USER Keyword Combination," on page 851
  - Appendix F, "Commands That Are Valid in ETO," on page 853
  - Appendix G, "Status and Attributes for the /DISPLAY Command," on page 855
  - Appendix H, "High Availability Large Database Commands," on page 867
  - Appendix I, "IMS Commands, RACF Access Authorities and Resource Names Table," on page 869

© Copyright IBM Corp. 1974, 2003

For quick reference to the commands, see *IMS Version 9: Summary of Operator Commands*, which contains syntax diagrams of all the commands described in this publication.

Prerequisite Knowledge

Before using this book, you should understand basic IMS concepts and your installation's IMS system. IMS can run in the following environments: DB batch, TM batch, DB/DC, DBCTL, and DCCTL. You should understand the environments that apply to your installation. For a complete list of courses, see the IMS home page on the IMS Web site at www.ibm.com/ims.

Recommendation: Before using this book, you should be familiar with the following books:

- *IMS Version 9: Operations Guide*
- *OS/390 JES2 Commands*
- *OS/390 JES3 Commands*
- *OS/390 MVS System Commands*

Syntax Diagrams

The following rules apply to the syntax diagrams used in this book:

Arrow symbols

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

►►— Indicates the beginning of a statement.

—► Indicates that the statement syntax is continued on the next line.

►— Indicates that a statement is continued from the previous line.

—►◄ Indicates the end of a statement.

Diagrams of syntactical units other than complete statements start with the ►— symbol and end with the —► symbol.

Conventions

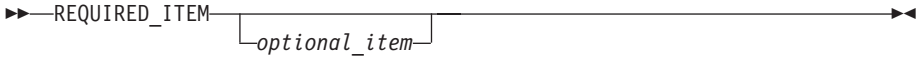
- Keywords, their allowable synonyms, and reserved parameters, appear in uppercase for MVS and OS/2 operating systems, and lowercase for UNIX operating systems. These items must be entered exactly as shown.
- Variables appear in lowercase italics (for example, *column-name*). They represent user-defined parameters or suboptions.
- When entering commands, separate parameters and keywords by at least one blank if there is no intervening punctuation.
- Enter punctuation marks (slashes, commas, periods, parentheses, quotation marks, equal signs) and numbers exactly as given.
- Footnotes are shown by a number in parentheses, for example, (1).
- A ␣ symbol indicates one blank position.

Required items

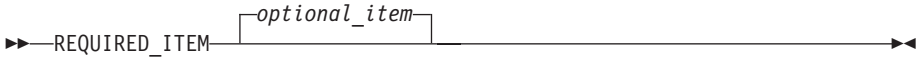
Required items appear on the horizontal line (the main path).

Optional Items

Optional items appear below the main path.

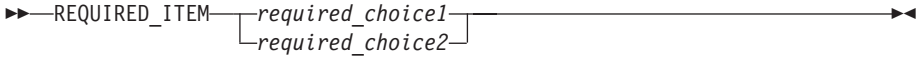


If an optional item appears above the main path, that item has no effect on the execution of the statement and is used only for readability.

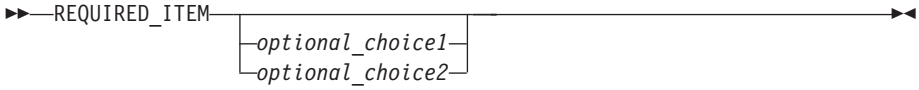


Multiple required or optional items

If you can choose from two or more items, they appear vertically in a stack. If you *must* choose one of the items, one item of the stack appears on the main path.

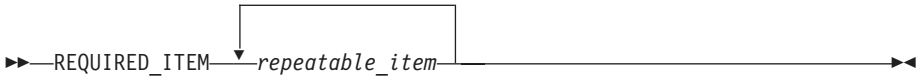


If choosing one of the items is optional, the entire stack appears below the main path.

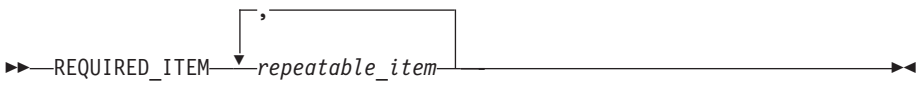


Repeatable items

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



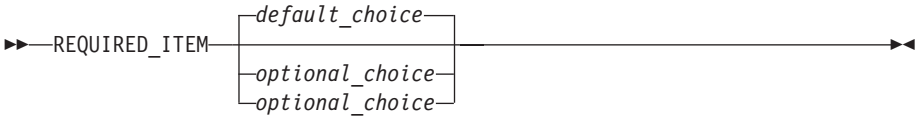
If the repeat arrow contains a comma, you must separate repeated items with a comma.



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

Default keywords

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

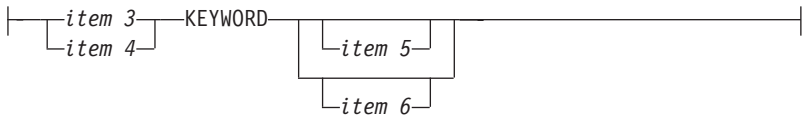


IMS-specific syntax information

Fragments

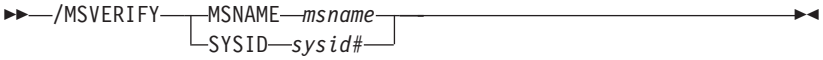
Sometimes a diagram must be split into fragments. The fragments are represented by a letter or fragment name, set off like this: | A |. The fragment follows the end of the main diagram. The following example shows the use of a fragment.

A:



Parameter endings

Parameters with number values end with the symbol '#', parameters that are names end with 'name', and parameters that can be generic end with '*'.



The MSNAME keyword in the example supports a name value and the SYSID keyword supports a number value.

How to Send Your Comments

Your feedback is important in helping us provide the most accurate and highest quality information. If you have any comments about this book or any other IMS documentation, you can do one of the following:

- Go to the IMS home page at: www.ibm.com/ims. There you will find an online feedback page where you can enter and submit comments.
- Send your comments by e-mail to imspubs@us.ibm.com. Be sure to include the name of the book, the part number of the book, the version of IMS, and, if applicable, the specific location of the text you are commenting on (for example, a page number or table number).
- Fill out one of the forms at the back of this book and return it by mail, by fax, or by giving it to an IBM representative.

Change Indicators

Technical changes are indicated in this publication by a vertical bar (|) to the left of the changed text.

Summary of Changes

Changes to This Book for IMS Version 9

New and Changed Commands and Keywords

Table 1 shows the new and changed commands and keywords for IMS Version 9. It also includes a brief description of the changes.

Table 1. Command and Keyword Changes for IMS Version 9

Command	Keyword	Description of Change
/DIAGNOSE	SNAP, BLOCK, NODE, TRAN	New command and keywords
QUERY	AREA, DB, NAME, OPTION, SHOW, STATUS	New keywords
/RECOVER	VERIFY	New keyword
UPDATE AREA	AREA, DATAGRP, DB, NAME, OPTION, SCOPE, SET, START, STOP	New keywords

This edition is a draft version of this book intended for use during the Quality Partnership Program (QPP). Contents of this book are preliminary and under development.

Library Changes for IMS Version 9

Changes to the IMS Library for Version 9 include the addition of new titles,the change of one title, and organizational changes.

New and Revised Titles

The following list details major changes to the IMS Version 9 library:

- *IMS Version 9: HALDB Online Reorganization Guide and Reference*

  The library includes a new book: *IMS Version 9: HALDB Online Reorganization Guide and Reference* (OLR). As a new book, the OLR is available only in PDF and BookManager.
- *IMS Version 9: An Introduction to IMS*

  The library includes a new book: *IMS Version 9: An Introduction to IMS* (INTRO).
- The book formerly titled *IMS Version 8: IMS Java User's Guide* is now titled *IMS Version 9: IMS Java Guide and Reference*.

Organizational Changes

TBD

Accessibility Enhancements

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use software products successfully. The major accessibility features in z/OS products, including IMS, enable users to:

- Use assistive technologies such as screen-readers and screen magnifier software
- Operate specific or equivalent features using only the keyboard

© Copyright IBM Corp. 1974, 2003

- Customize display attributes such as color, contrast, and font size

User Assistive Technologies

Assistive technology products, such as screen readers, function with the user interfaces found in IMS. Consult the assistive technology documentation for specific information when using it to access these interfaces.

Accessible Documentation

Online documentation for IMS Version 9 is available in BookManager format, which is an accessible format. All BookManager functions can be accessed by using a keyboard or keyboard shortcut keys. BookManager also allows you to use screen readers and other assistive technologies. The BookManager READ/MVS product is included with the OS/390 base product, and the BookManager Library Reader (for workstations) is available on the IMS Licensed Product Kit (CD), which is available for downloading from IBM at www.ibm.com.

Keyboard Navigation of the User Interface

Users can access IMS user interfaces using TSO/E or ISPF. Refer to the *z/OS : TSO/E Primer*, *z/OS : TSO/E User's Guide* , *z/OS : ISPF User's Guide*. These guides describe how to navigate each interface, including the use of keyboard shortcuts or function keys (PF keys). Each guide includes the default settings for the PF keys and explains how to modify their functions.

Part 1. Introduction

Chapter 1. IMS Command Language . . . 3
Introduction . . . 3
How to Enter an IMS Command . . . 3
IMS MFS 3270 Master Terminal Format . . . 3
Using Multiple Console Support (MCS) Consoles . . . 6
Outstanding Reply Numbers on z/OS Consoles . . . 7
Maximum Length of Command Input from z/OS Consoles . . . 8
Multisegment Command Input . . . 8
Commands in a DBCTL Environment . . . 9
Commands from an LU 6.2 Device . . . 10
Qualifying Network LU Names and Commands . . . 10
TSO SPOC Format . . . 11
3 Control Center Format . . . 12
Commands from the OM API . . . 12
Command Responses to OM . . . 13
Command Responses . . . 14
Responses to LU 6.2 Devices . . . 14
Sending Messages to the z/OS System Console . . . 15
Sending Messages to the IMS Master Terminal . . . 15
IMS Terminal Command Examples . . . 15
TSO SPOC Command Examples . . . 16
OM API Command Examples . . . 16
IMS Classic Command Format . . . 16
Command Recognition Character . . . 17
Verb . . . 17
Keywords . . . 17
Parameters . . . 17
Null Keywords . . . 20
Comments . . . 21
3 Passwords . . . 21
Command Processing in an IMSplex . . . 21
Command Routing . . . 21
Command Behavior and the Resource Manager . . . 22
IMSplex Command Format . . . 22
Command Characteristics . . . 24
Terminal Security Defaults . . . 24
Commands Recovered During Emergency Restart . . . 26
Commands Logged to the Secondary Master Terminal . . . 27
Commands Supported from an AO Application . . . 27
Using CMD Call . . . 28
Using ICMD Call . . . 30
Command Security When Using OM . . . 33
Commands Mirrored on an XRF Alternate . . . 33
Commands Supported on the XRF Alternate . . . 34
Commands and Keywords Supported on an RSR Tracking Subsystem . . . 36
BTAM Terminals in an RSR Environment . . . 38
Commands Supported from LU 6.2 Devices and OTMA . . . 38
Terminal Security Defaults for LU 6.2 Devices and OTMA . . . 39
Commands Supported by the OM API . . . 39

Chapter 2. Keyword Table and Definitions . . . 43

© Copyright IBM Corp. 1974, 2003

Chapter 1. IMS Command Language

Introduction

This chapter describes how to use the IMS™ master terminal, TSO SPOC, or the OM API for messages, commands, and responses; how to input IMS classic commands and IMSplex commands; and how to read the command syntax diagrams, command characteristics, and command keyword definitions. The commands are described in alphabetical order in Part 2, “IMS Commands,” on page 79

Classic IMS commands can be entered from the master terminal. IMS classic commands are commands that are in the original IMS command syntax. Certain classic IMS commands can be entered from remote terminals, from the master terminal, or both. The criteria used for determining which commands can be entered from which terminals are established differently for statically defined terminals and for dynamically created terminals. IMS system definition and the Security Maintenance utility establish the criteria for statically defined terminals. RACF® or equivalent security product establishes the criteria for dynamically created terminals.

The standard terminal command defaults created by system definition can be modified by the Security Maintenance utility. Table 3 on page 24 shows the terminal security defaults for IMS commands established by IMS system definition.

Optionally, commands can be entered from a TSO SPOC or the OM API. Both IMS classic commands and IMSplex commands can be entered from a TSO SPOC or the OM API. However, IMSplex commands can only be entered from an OM API, not a master or remote terminal. Some classic IMS commands are not supported by the OM API and some are not recoverable.

z/OS commands can be entered only at the z/OS system console and not from the IMS master terminal.

How to Enter an IMS Command

This section describes the following:

- “IMS MFS 3270 Master Terminal Format”
- “Using Multiple Console Support (MCS) Consoles” on page 6
- “Outstanding Reply Numbers on z/OS Consoles” on page 7
- “Maximum Length of Command Input from z/OS Consoles” on page 8
- “Multisegment Command Input” on page 8
- “Commands in a DBCTL Environment” on page 9
- “Commands from an LU 6.2 Device” on page 10
- “Qualifying Network LU Names and Commands” on page 10
- “TSO SPOC Format” on page 11
- “Control Center Format” on page 12

IMS MFS 3270 Master Terminal Format

This section describes how to use the IMS master terminal for messages, commands, and responses. The IMS master terminal consists of two components of the IBM® 3270 Information Display System:

© Copyright IBM Corp. 1974, 2003

- A 3270 display screen with 24 lines by 80 columns; a screen size of 1920 characters
- A 3270 printer

All IMS system messages for the master terminal are sent to the display screen. Most system messages, commands and responses are also sent to the printer to provide a hard copy record.

The format of the display screen is normally divided into five areas:

- System status area (line 1)
- Message area (lines 2-10)
- Display area (lines 12-21)
- Warning message area (line 22)
- User input area (lines 23 and 24)

Figure 1 is a screen that shows the format of the master terminal display screen.

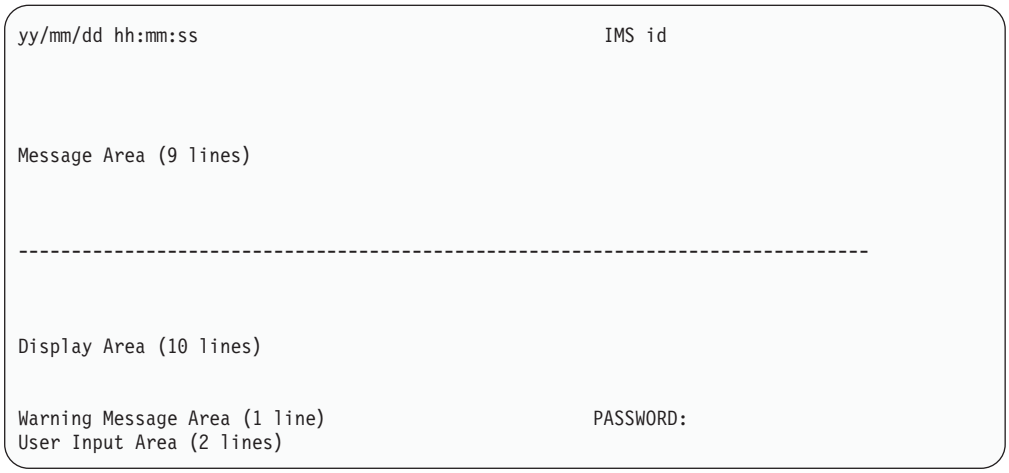


Figure 1. Master Terminal Format

Figure 2 on page 5 is a screen that shows the format of the master terminal display screen in an extended recovery facility (XRF) environment.

```
85/05/15 14:24:40  RSENAME: DFSRSENM   BACKUP  TRACKING IN PROGRESS   IMSA


Message Area (9 lines)


--------------------------------------------------------------------------------


Display Area (10 lines)



Warning Message Area (1 line)                          PASSWORD:
User Input Area (2 lines)
```

Figure 2. Master Terminal Format in XRF Environment

System Status Area

This area (line 1) displays the date, time, and IMS identifier:

Date Current date

Time Current time

IMS Identifier
IMS system ID

If the system is XRF capable, the following information is also displayed on the status line:

RSEname Recoverable service element name

System ACTIVE or BACKUP (field is completed following the initialization phase)

Phase Blank or one of the following:

Initialization Active or alternate system is in initialization phase

Synchronization
Alternate system processing SNAPQ checkpoint

Tracking in progress
Alternate system synchronized with active system and in the process of tracking active system

Takeover in progress
Alternate system in the process of taking over from active system

Awaiting I/O prevention
New active system is in I/O toleration mode and is awaiting completion of I/O prevention on the failing system prior to becoming a true active system

Message Area

This area (lines 2 through 10) displays IMS system messages, messages from other terminals and programs, and responses to IMS commands, except responses to a /DISPLAY command. If the message area is full and more output needs to be

displayed, a warning message is sent. Press the PA1 key to receive more output. New data is displayed on the top line followed by a blank line to separate new data from old messages.

Display Area

This area (lines 12 through 21) displays the output from a /DISPLAY command. If the response to the command exceeds the size of the area, a warning message is sent.

Warning Message Area

This area (line 22) can contain one of the following warning messages:

MASTER LINES WAITING
A message being displayed in the message area has exceeded the available space and is only partially displayed.

MASTER MESSAGE WAITING
The message area is full, and a message from another terminal or program is waiting to be displayed.

DISPLAY LINES WAITING
The output of a /DISPLAY command exceeds the size of the display area.

USER MESSAGE WAITING
A message requiring the screen to be formatted differently is waiting to be displayed.

In all cases, press the PA1 key to receive the waiting output.

The literal PASSWORD is followed by an unprotected field with the nondisplay attribute at the end of line 22.

User Input Area

This area (lines 23 and 24) is used to enter IMS commands. It consists of two 79-character segments, so you can enter multisegment commands such as /BROADCAST, /CHANGE, /RMxxxxxx, and /SSR. For example, to send a message to all active terminals, enter /BROADCAST ACTIVE on line 23 and enter the message text on line 24. The cursor is initially positioned by IMS to line 23, your first input line.

Sometimes the screen can lose this special format (for example, if you have pressed PA1 and received a user message, or if you have pressed the Clear key). To restore the format, enter /FORMAT DFSM02. If the screen is currently formatted with an application display, this command can be entered on the command input line at the bottom of the screen and must be terminated with a period.

Using Multiple Console Support (MCS) Consoles

You can enter IMS commands from Multiple Console Support (MCS) or Extended Multiple Console Support (E-MCS) consoles. Use the CMDMCS keyword for the IMS, DBC, or DCC procedures during IMS system definition to allow commands to be entered from MCS or E-MCS consoles. See *IMS Version 9: Installation Volume 2: System Definition and Tailoring* for more information on the CMDMCS keyword.

In a DB/DC environment, if Multiple Console is enabled (CMDMCS other than 'n' specified), IMS commands can be entered from an MCS or E-MCS console by prefixing the command by either the CRC or IMSID. For example, if the CRC is '#' then a valid command would be #DIS A. If the IMSID is 'IMSP' then a valid command would be IMSPDIS A.

Each segment of an IMS command must begin with one of the following:

- The command-recognition character for the IMS subsystem
- The IMS ID of the IMS subsystem

For multisegment commands, each segment, except the last, must end with the command recognition character or IMS ID followed by the ENTER key. The last segment must end with only the EOM (end-of-message, or ENTER) character.

Prior to IMS Version 9, the only method of sending a command to all IMS subsystems in a sysplex (hereafter called an IMSplex) was to issue the z/OS `ROUTE *ALL` command from an MCS or E-MCS console. See *OS/390 MVS System Commands* for more information on the `ROUTE *ALL` command.

As of IMS Version 9, you can send commands to all the IMSs in an IMSplex by using an automated operations single-point-of-control (SPOC) application that sends commands and receives the responses to those commands through the Operations Manager (OM) application programming interface (API).

Definition: An IMSplex is one or more IMS subsystems that work together in groups that share databases, message queues, or both. Additionally, for the purposes of this book, this definition includes the idea of these IMSs being managed by components of the Common Service Layer (CSL), the Operations Manager (OM), the Resource Manager (RM), and the Structured Call Interface (SCI).

Related Reading: For more information about managing multiple IMSs in an IMSplex using SPOC application programs, see “Automated Operations” in the *IMS Version 9: Operations Guide*. For more information about the CSL components and the OM API, see the *IMS Version 9: Common Service Layer Guide and Reference*.

When you are using a Resource Manager (RM) structure in an IMSplex, it is best to issue commands to the Operations Manager (OM) through the TSO SPOC.

Outstanding Reply Numbers on z/OS Consoles

Outstanding reply numbers are used only on z/OS system consoles. They are used to pass input directly to IMS. Terminals defined to IMS do not use outstanding reply numbers.

As an example, IMS might display the following on the z/OS console:

```
*48 DFS996I *IMS READY*
```

To communicate with IMS through the z/OS system console, you must reply with the outstanding reply number (in the above example, the number is `48`). A reply might look like this:

```
R 48,/START DC
```

After IMS responds to the command, the next outstanding reply number is displayed with the DFS996I message, as follows:

```
DFS058I START COMMAND COMPLETED
*49 DFS996I *IMS READY*
```

Maximum Length of Command Input from z/OS Consoles

For single-segment commands that are entered from the z/OS console, the maximum character length (including command recognition characters), is 118. This limit is due to the current implementation of the WTOR macro.

Multisegment Command Input

Multisegment commands require an EOS (end-of-segment) indicator for all segments preceding the last segment, and an EOM (end-of-message) indicator for the last segment. EOS and EOM will vary depending on the type of terminal entering the command. For example, on the system console, EOS is indicated by the ENTER key and EOM is indicated by entering a period followed by the ENTER key. Refer to the *IMS Version 9: Operations Guide* for a discussion of multisegment input for various device types.

The following are multisegment commands:

- /ALLOCATE
- /BROADCAST
- /CHANGE
- /ERESTART
- /LOOPTEST
- /OPNDST
- /RMxxxxxx
- /SSR

The period is used both as a delimiter for comments on IMS commands and as an EOM indicator for some terminal types. The following examples illustrate the use of the period in both contexts. “Example 4 for Multisegment Commands” on page 9 shows this.

Example 1 for Multisegment Commands

System console entry:

```
24,/BRO MASTER. END OF DAY MESSAGE
25,I AM GOING HOME.  PLEASE TAKE OVER.
```

Response on master terminal:

```
I AM GOING HOME.  PLEASE TAKE OVER.
```

Explanation: The first input segment (reply 24) contains only the command and consists of a comment (END OF DAY MESSAGE) separated from the command itself (BRO MASTER) by a period. The second input segment (reply 25) consists of the data to be broadcast. The first period in segment 2 is part of the data and the second period in segment 2 is the EOM indicator for the system console.

Example 2 for Multisegment Commands

System console entry:

```
26,/CHANGE DIRECT
27,ORY MFS. MODIFICATION.
```

Explanation: MODIFICATION in segment 2 is a comment and is separated from the command by the first period. The second period in segment 2 is the EOM indicator for the system console.

The above 2 segments are combined and the following is input to the command processor:

```
/CHANGE DIRECTORY MFS. MODIFICATION
```

Example 3 for Multisegment Commands

System console entry:

```
28,/CHANGE LINK 1
29,2 3 FORCSESS.
```

Explanation: Certain forms of reply to the system console “outstanding reply” can compress out leading blanks from input segments. In the above example, the command processed by the /CHANGE command processor would be: /CHANGE LINK 12 3 FORCSESS. To obtain the desired command, /CHANGE LINK 1 2 3 FORCSESS, one of the following forms of reply could be used:

```
R 29,'/CHANGE LINK 1'
R 30,' 2 3 FORCSESS.'
```

or

```
29,/CHANGE LINK 1
30, 2 3 FORCSESS
```

Example 4 for Multisegment Commands

LU2 or 3270 Display Terminal entry:

```
/FORMAT DFSM04
/BRO LTERM WTOR (eos)
this is segment 1 (eos)
this is segment 2 (eos)
this is segment 3 (eom)
```

Explanation: The remote terminal is first formatted by the/FORMAT command, where default format DFSMO4 supports the input of 4 segments.

For LU2 and 3270 display terminals, enter a /FORMAT DFSM04 command before entering the multisegment command.

Commands in a DBCTL Environment

In a DBCTL environment, there are no outstanding reply numbers for z/OS consoles, and therefore none should be entered. For a command, just enter the command recognition character (slash or other character preceding the verb) or IMS ID, followed by the command. DBCTL does not respond with the DFS996I outstanding reply number.

Examples for DBCTL Commands

System console entry for single-segment command:

```
¢DIS DB PAYROLL
```

Explanation: ¢ is the command-recognition character for the DBCTL subsystem.

```
IMS1DIS DB PAYROLL
```

Explanation: IMS1 is the IMS ID for the DBCTL subsystem.

System console entry for multisegment command:

```
/RMI DBRC='IC DBD(DEDBDD01) AREA (DD01AR0) ICDSN(FVT31.DEDBDD01.DD01
AR0.IC.DUMMY1) ICDSN2/
/(FVT31.DEDBDD01.DD01AR0.IC2.DUMMY1) HSSP'
```

Explanation: The first segment of the /RMI command has the command recognition character at the beginning of the segment and at the end of the segment, to indicate another segment follows. The last segment has the command recognition character at the beginning of the segment only.

Commands from an LU 6.2 Device

To enter a command from an LU 6.2 device, allocate a conversation using an IMS command verb as the TPNAME. The "/" must be included in the command word. The password option is not valid. (The password must be supplied as a part of the LU 6.2 ALLOCATE from the partner.) The normal LU 6.2 security provisions apply for this transaction.

Restriction: The APPC synchronization level of CONFIRM is ignored for commands from LU 6.2 devices.

Following are examples of the sequence of steps used to issue commands by allocating a synchronous conversation on an LU 6.2 device:

Example 1 for Commands from an LU 6.2 Device

```
ALLOCATE
  -LUNAME=luname of IMS
  -TPNAME=/DISPLAY                    (Single segment command)
SEND
  -DATA=LTERM VA01 VA02.comments      (Data after period treated as comments)
PREPARE_TO_RECEIVE
RECEIVE                               (Get header)
RECEIVE                               (Get output for LTERM VA01)
RECEIVE                               (Get output for LTERM VA02)
RECEIVE                               (Get timestamp)
RC=DEALLOCATE
```

Example 2 for Commands from an LU 6.2 Device

```
ALLOCATE
  -LUNAME=luname of IMS
  -TPNAME=/CHANGE                     (Multisegment command)
SEND
  -DATA=NODE                          (Can spread over several SENDs)
SEND
  -DATA=VAT02 ASR OFF.comments        (Data after period treated as comments)
PREPARE_TO_RECEIVE
RECEIVE                               (Get DFS058 CHANGE COMMAND COMPLETED)
RC=DEALLOCATE
```

Example 3 for Commands from an LU 6.2 Device

```
ALLOCATE
  -LUNAME=luname of IMS
  -TPNAME=/BROADCAST
SEND
  -DATA=NODE VAT0*                    (1st SEND: the destination)
SEND
  -DATA=HELLO, HOW ARE YOU ALL?       (Subsequent SENDs: message to broadcast)
PREPARE_TO_RECEIVE
RECEIVE                               (Get DFS058 BROADCAST COMMAND COMPLETED)
RC=DEALLOCATE
```

Qualifying Network LU Names and Commands

Qualifying LU names gives the system administrator the liberty of using the same name for LUs on different systems by adding the network identifier. This eliminates the necessity to have unique names for every LU on every system in your complex.

A network-qualified LU name consists of a network identifier of the originating system that is 1 to 8 characters long, a period, and then the LU name, which is 1 to 8 characters long. IMS commands that include network-qualified LU names must be enclosed in single quotes. For example: `'NETID001.LUAPPC02'`. No blanks are allowed in a network-qualified LU name.

The parameter ALL for either the network identifier or the LU name cannot be substituted in a command for a network-qualified LU name. ALL cannot represent all of the LU names or all of the networks.

The LU name in the LU 6.2 descriptors can be network-qualified.

The network-qualified LU name is optional on commands that support the LUNAME keyword.

TSO SPOC Format

The TSO SPOC is an IBM-supplied application which can issue operator commands in an IMSplex. The TSO SPOC application uses an ISPF panel interface and communicates with an Operations Manager (OM) address space. OM then communicates with all of the other address spaces in the IMSplex (for example, IMS) as required for operations.

There can be more than one TSO SPOC in an IMSplex. However, it is optional in an IMSplex. The TSO SPOC provides the following functions to an IMSplex:

- Presents a single system image for an IMSplex by allowing the user to enter commands to all IMSs in the IMSplex from a single console.
- Displays consolidated command responses from multiple IMS address spaces.
- Sends a message to an IMS terminal connected to any IMS control region in the IMSplex by using the IMS `/BROADCAST` command.

The TSO SPOC has integrated online help which can be accessed by the PF1 key. There are three ways to issue commands in the IMS TSO SPOC application:

- By command line
- By retrieving a command
  - Using the ISPF RETRIEVE command
  - Using a command listed in the response area
  - Using the Command status panel
- By defining and using command shortcuts

These three methods can be used in any combination at any time. See the IMS TSO SPOC Online Tutorial, for more information about using the TSO SPOC application. For more IMSplex information, see *IMS Version 9: Common Service Layer Guide and Reference*.

Figure 3 on page 12 is a screen that shows the format of the TSO SPOC screen.

```
File     Display    View    Options    Help
-------------------------------------------------------
PLEX1                          IMS Single Point of Control
Command ==> ___________


------------------------- Plex .    Route .     Wait .
Response for:


CSLM000I (C) Copyright IBM Corp. 2000. All rights reserved.
F1=Help     f3=Exit    F4=Showlog    F6=Expand     F9=Retrieve     F12=Cancel
```

Figure 3. TSO SPOC Screen Format

You can enter both IMS classic commands and IMSplex commands using the TSO SPOC interface. The command is entered next to the command prompt (`Command ==>` in the figure). Enter the IMSplex name in the `Plex` field. Enter the list of IMSs to which to route the command, if applicable, in the `Route` field. After you type the command, press Enter. The command issued is shown in the `Response for:` field and the actual command response is shown below the `Response for:` field.

Entering Comments from a TSO SPOC

For classic IMS commands, a period ends a command and anything after it is a comment. The following example shows a classic IMS command with a comment.

```
/DISPLAY ACT .a test comment
```

For IMSplex commands, the comment must be enclosed in a slash and asterisk. At the end of the last parameter of the command, type a forward slash followed by an asterisk, then the comment text, followed by an asterisk and slash. The comment text must not have an embedded asterisk slash in it. The following is an example of a command with a comment:

```
QUERY TRAN NAME(PART) SHOW(QCNT)/*Show queue count for tran part*/
```

Control Center Format

The IMS Control Center, included in the IBM DB2 Universal Database Control Center, uses the IMS Single Point of Control (SPOC) functions to enable IMS systems management from a Windows workstation.

You can issue and view IMS type-1 and type-2 commands from the IMS Control Center. There are online wizards that help you build and issue commands. The Control Center command output is similar to the output displayed from the TSO SPOC. The same information is provided, but formatted in a windows-based graphical interface.

For more information about the IMS Control Center, go to the IMS website at www.ibm.com/ims and link to the IMS Control Center.

Commands from the OM API

An IMSplex is one or more IMS subsystems that work together in groups that share databases, message queues, or both, and are managed by components of the Common Service Layer (CSL). The CSL includes the Operation Manager (OM) API. For more information about the CSL, see *IMS Version 9: Common Service Layer Guide and Reference*. IMSplex commands, or commands from the OM API work differently from classic IMS commands. Classic IMS commands are those that are

in the original or ″classic″ IMS command syntax--the command format allowed from the IMS master terminal. Following are the general differences between IMSplex commands and classic IMS commands:

- IMSplex commands are commands that apply to the entire IMSplex. They can be entered only from the OM API.
- Some classic IMS commands are not supported by the OM API. For a list of commands supported by the OM API, see “Commands Supported by the OM API” on page 39.
- Some classic IMS commands are not recoverable. For a list of recoverable commands, see Table 4 on page 26.

In this book, IMSplex commands are found in alphabetical order in Part 2, “IMS Commands,” on page 79 In this section, the IMSplex commands are those without a slash or command recognition character. For more information about the IMSplex command format, see “IMSplex Command Format” on page 22.

Command Responses to OM

When an IMSplex command or IMS classic command is issued from the OM API, OM routes it to one or more IMS systems based on the command routing. Each IMS returns the command response to OM. The command responses from each IMS are grouped together by OM and returned to the caller.

For each IMS command that is issued from OM API, the response is encapsulated in XML tags. For a list of XML tags used, an explanation of the format, and meaning of the tags, see *IMS Version 9: Common Service Layer Guide and Reference*.

The IMSplex command responses to OM also include a return code, reason code, and a possible completion code. The OM return and reason codes can be found in the ’CSLOMCMD Return and Reason Codes’ table in the *IMS Version 9: Common Service Layer Guide and Reference*.

When the following classic commands are issued from OM API, the DFS058I COMMAND IN PROGRESS message is not returned. Instead, the messages that result as a part of command processing are returned to OM. Following is the list of commands that do not return the DFS058I COMMAND IN PROGRESS:

- `/DBDUMP DATABASE`
- `/DBRECOVERY AREA`
- `/DBRECOVERY DATABASE`
- `/DBRECOVERY DATAGROUP`
- `/LOCK DATABASE`
- `/START AREA`
- `/START DATABASE`
- `/START DATAGROUP`
- `/STOP ADS`
- `/STOP AREA`
- `/STOP DATABASE`
- `/STOP DATAGROUP`
- `/UNLOCK DATABASE`
- `/VUNLOAD AREA`

Command Responses

With the exception of the /DISPLAY command, /FORMAT command, and type-2 commands returned through the OM API, responses to IMS commands are prefixed by the letters DFSnnn; nnn identifies the message. For information about the messages, see *IMS Version 9: Messages and Codes, Volume 1*. A response to an IMS command is an acknowledgment from IMS to the originating terminal that the command has been received. Responses that go to the system console have an IMS ID identifying the IMS system that issued the message. For example, the response to /DBDUMP DATABASE MSDB would be:

```
DFS058I (time stamp) DBDUMP COMMAND IN PROGRESS  (IMS id)
```

At system definition, the TIMESTAMP/NOTIMESTP. parameter of the COMM macro determines whether the time stamp is present or absent. If the time stamp feature is included, the date and time of the response appear between the response prefix and the text.

The DFS058 COMMAND COMPLETED/IN PROGRESS response indicates whether IMS accepted the command. If some parameters of the command are not accepted, the response includes the EXCEPT phrase and indicates the parameters that were not accepted. If IMS does not have the space for all of the not-accepted parameters, it truncates the EXCEPT phrase and terminates it with ...etc.. Commands that specify the ALL parameter are most likely to be truncated. Truncated EXCEPT phrases on commands are normally caused by:

- Misspelling a parameter
- Specifying an invalid parameter for the command
- Specifying the ALL parameter for resources when some of them are already in the requested state

Responses to LU 6.2 Devices

The response to a command from an LU 6.2 device is sent synchronously if the initiating conversation is not deallocated by the remote device. If the conversation is deallocated, the command response is sent asynchronously using the LUNAME of the device and a TPN of DFSCMD. The remote device must allocate separate DFSCMD conversations to receive each asynchronous message.

When the following commands are entered from LU 6.2 devices:

- /DBDUMP DATABASE
- /DBRECOVERY AREA
- /DBRECOVERY DATABASE
- /START AREA
- /START DATABASE
- /START REGION
- /STOP AREA
- /STOP DATABASE
- /STOP REGION

the DFS058 COMMAND COMPLETED message is generated instead of the DFS058 COMMAND IN PROGRESS message. If the LU 6.2 conversation is synchronous (the conversation still exists), the DFS058 COMMAND COMPLETED message is the last message the operator receives. If the LU 6.2 conversation is asynchronous (the conversation is deallocated), the order of delivery of the DFS058 message cannot be guaranteed.

Sending Messages to the z/OS System Console

Each z/OS system console in the sysplex has a unique LTERM name. In an XRF environment, the real z/OS system console name is the RSE name specified in the HSB member. For a non-XRF subsystem, the real system console name is the IMS ID. The generic z/OS system console name is always “WTOR”. The generic name is translated to the real LTERM name when a message is sent to the z/OS system console.

Because each z/OS system console has a unique LTERM name, a message can be sent to any z/OS system console in the sysplex if the real LTERM name (not the generic name) is used.

Sending Messages to the IMS Master Terminal

Each IMS master terminal (MTO) in the sysplex has a unique LTERM name, and if they are VTAM® terminals they also have unique node names. Each IMS assigns its primary MTO the default LTERM name “DFSPMTO” as a generic name. Each secondary MTO is assigned “DFSSMTO” as the default generic name. You can override either of these default generic names during IMS system definition by using the DFSDCxxx member of IMS.PROCLIB. The generic name is translated to the real LTERM name when a message is sent to the MTO.

Restrictions: If you use the DFSDCxxx member to override a default generic name, you must not specify a statically-defined node name. Also, the node name you specify must have the same physical characteristics as the default node you are overriding. If you change the DFSDCxxx member, you must perform a cold start of IMS. See *IMS Version 9: Installation Volume 2: System Definition and Tailoring* for more information on this member.

In a shared-queues environment, because each MTO has a unique LTERM name, a message can be sent to any MTO in the sysplex if you use the real LTERM name (not the generic name).

IMS Terminal Command Examples

Some command examples show the format of commands entered and received from an IMS terminal--entry terminal (ET), master terminal (MT), or response terminal (RT). One or more examples are given for each command. The examples show:

Entry ET:

```
How the command looks as entered from the entering
terminal (ET).
```

Response ET:

```
The IMS response transmitted to the entering terminal (ET).
```

Entry MT:

```
How the command looks as entered from the master
terminal (MT).
```

Many IMS commands reference other terminals. Examples of these commands include:

Response RT:

```
The IMS response transmitted to the referenced terminal or
terminals (RT).
```

The ET and RT responses might or might not print or display, depending on what type terminal is referenced and whether the MSGDEL (message delete) option was specified for the terminal at system definition time.

TSO SPOC Command Examples

Some command examples show the input and output of commands from a TSO SPOC. Both IMSplex commands and classic commands can be entered from a TSO SPOC. One or more examples are given for each command. The examples show:

TSO SPOC input:

```
How the command looks as entered from the TSO SPOC.
```

TSO SPOC output:

```
The command response as displayed by the TSO SPOC.
```

Explanation: An explanation of the command and response.

Some IMSplex commands include a section describing the environment of the IMSplex prior to entering the command. A response panel, log panel, or both displays when a command is entered from a TSO SPOC. Output data from the response panels are included in the examples, but output data from the log panels are not. The log panels display the OM return and reason codes for the command. For information about the OM return and reason codes, the list of codes can be found in the 'CSLOMCMD Return and Reason Codes' table in the *IMS Version 9: Common Service Layer Guide and Reference*.

OM API Command Examples

Some command examples show the input and output of commands to and from the OM API. Both IMSplex commands and classic commands can be entered from the OM API. One or more examples are given for each command. The examples show:

OM API input:

```
How the string looks as built by the AOP client.
```

OM API output:

```
The XML response transmitted from OM.
```

Explanation: An explanation of the command and response.

Some IMSplex commands include a section describing the environment of the IMSplex prior to entering the command.

IMS Classic Command Format

The general format of an IMS command is shown below.

Command Recognition Character

The slash (/) is the command recognition character for IMS commands. In a DBCTL environment, another character can be defined as the command recognition character and would replace the slash.

Verb

The /VERB (such as /STOP) is the first command element and identifies the action to be taken. To abbreviate a verb, use only the slash and the first three characters. Enter the characters using either uppercase or lowercase letters.

Keywords

IMS keywords are reserved words identifying IMS resources that relate to transaction management, databases, or the IMS system. Many commands require one or more keywords.

Table 14 on page 43 is a list of available keywords and their synonyms. The alphabetical list of keywords after Table 14 on page 43 explains the use of the keyword for each IMS command it affects.

Restriction: Keywords and their synonyms are reserved words. Do not use reserved words to name system resources such as transactions, programs, logical terminals, databases, and MFS block names.

Keywords are shown in uppercase. The keyword or an allowable synonym can be entered in uppercase or lowercase.

Parameters

Parameters refer to installation defined or created resources, such as nodes, databases, or pterms, or IMS reserved parameters. Parameters in IMS commands must be replaced with parameter values. Reserved parameters have a unique meaning to IMS and refer to system resources or functions.

The format of the parameters that can be entered is determined by the command and keyword with which they are associated. Use the syntax example provided at the beginning of each command description in Part 2, “IMS Commands,” on page 79 to determine the valid parameter formats for that command.

When commands are entered, parameters are separated from a keyword by a space or an equal sign (=). Multiple parameters are separated by a comma, a blank, or a comma followed by a blank. The last parameter of one keyword is separated from the next keyword either by a space, a comma, or an equal sign.

Default parameters appear above the syntax line. Optional parameters appear below the syntax line.

In the syntax examples, a repeat-line is shown as follows:

The repeat-line indicates that multiple parameters can be entered. To save time while entering multiple parameters for the LINE, PTERM, NODE, and LINK keywords, inclusive parameters can be specified in some commands.

Restriction: Use of the ALL parameter, generic parameters, or excessive use of multiple or inclusive parameters in the large terminal network can cause some of the following problems:

- "Lock out" of the operator for the duration of the command processing
- Creation of large multisegment output messages
- Command action for only part of the terminal resources intended
- Temporary storage shortages
- Artificially high storage pools within IMS, VTAM, or both

Reserved Words

The resources you define, such as nodes, pterms, and databases, may be referenced as parameter values in IMS commands. Do not use reserved words to name these resources, or IMS may not interpret the command as expected. For example, if you define a node called TKOTRA and issue a `/DISPLAY NODE TKOTRA` command, IMS will list all the nodes that have an attribute of TKOTRA, instead of the specific node named TKOTRA. Appendix C, "List of Reserved Words," on page 845 contains a table of the reserved words for IMS commands. These include all keywords, null words, attributes for various forms of the `/DISPLAY` command, and IMS reserved parameters OFF, MSDB, PRIMARY, NONE and ALL.

ALL Parameter

Entering the reserved parameter ALL specifies all defined resources related to that keyword. ALL cannot be used with every command.

ALL=NO or ALL=DIS can be specified on the KEYWD macro to prevent use of the ALL parameter. ALL=NO and ALL=DIS can be specified for all keywords. ALL=NO indicates that the ALL parameter is invalid on all the commands for the keyword. ALL=DIS indicates the ALL parameter is invalid on all the `/DISPLAY` commands for the keyword. For more information on the KEYWD macro, see *IMS Version 9: Customization Guide*.

Inclusive Parameters

Inclusive parameters are used to specify a group of parameters by entering only the first and last parameters of the group. Inclusive parameters can be used for the following resources:

- Line
- Link
- Static node
- PTERM

To use inclusive parameters, enter the first parameter, a hyphen, then the last parameter (parameter1-parameter2).

For the LINE, PTERM, and LINK keywords, the group of parameters specified is assumed to be based on numeric sequence; that is, entering 3-6 specifies parameters 3, 4, 5, and 6. Inclusive parameters are limited to the PTERM keyword when it appears with the LINE keyword in a command. For the NODE keyword, the group of parameters must be static and based on the IMS definition sequence, not numeric or alphabetic sequence.

Generic Parameters

Generic parameters allow easy specification of a group of resources, whose names match the mask supplied by the generic parameter.

A generic parameter is a 1- to 8-character alphanumeric name that includes one or more special characters and an asterisk or a percent sign. An asterisk can be replaced by zero or one or more characters to create a valid resource name. A percent sign can be replaced by exactly one character to create a valid resource name.

The generic parameter is * = ALL. However, depending on the installation, other generic parameters can mean ALL. For example, the generic parameter %%%% means ALL to an installation whose resource names are all four characters long.

When a generic parameter is used, IMS uses a serial search of all resources to match the generic name. Using a serial search of all resources can have performance implications in larger systems.

Restriction: If you specify ALL=NO on the KEYWD macro, you cannot use *. However, if you specify ALL=NO on the KEYWD macro, you can use other generic parameters that are equivalent to ALL. If you are using BASIC EDIT, and you enter a command with a generic parameter containing two asterisks in a row at the end of the command, the command is discarded.

Generic parameter wildcard characters for IMSplex commands are the same as for classic commands. An asterisk (*) in the name can be specified to replace one or more characters in the command string. A percent sign (%) replaces one character.

3 The commands listed in Table 2 support generic parameters for LTERM, MSNAME,
3 NODE, TRANSACTION, USER, USERID, NAME, and PROGRAM keywords.

Table 2. Commands and Keywords that Support Generic Parameters

Command	Keyword Supported
/BROADCAST	LTERM, NODE, USER
/CHANGE	NODE
/CLSDST	NODE
/DISPLAY	LTERM, MSNAME, NODE, TRANSACTION, USER
/DISPLAY ASMT	LTERM, MSNAME, NODE, USER
/DISPLAY CONV	NODE, USER
/IDLE	NODE
/OPNDST	NODE
/PSTOP	LTERM, TRANSACTION
/PURGE	LTERM, MSNAME, TRANSACTION
3 QUERY AREA	NAME
3 QUERY DB	NAME
3 QUERY LE	LTERM, TRAN, USERID, PROGRAM
3 QUERY TRAN	NAME
/RSTART	NODE, USER
/START	LTERM, MSNAME, NODE, TRANSACTION, USER
/STOP	LTERM, MSNAME, NODE, TRANSACTION, USER
/TRACE	NODE
3 UPDATE AREA	NAME
3 UPDATE DB	NAME

Table 2. Commands and Keywords that Support Generic Parameters (continued)

Command	Keyword Supported
UPDATE TRAN	NAME

Restriction: Generic parameters are not supported in NODE, USER, or TRANSACTION CLASS combinations.

Group Parameters

A group parameter allows easy specification of a group of resources, where the resources in the group are defined and managed by the installation. Group parameters are supported with the DATAGROUP keyword for the /DBRECOVERY, /START,/STOP, and UPDATE (type-2) commands. A group name is 1 to 8 characters long. It is associated with a list of databases, DEDB areas, or database data sets.

A database group name and its associated databases or DEDB areas is defined in the DBRC RECON data set using the DBRC INIT.DBDSGRP command with parameters GRPNAME and DBGRP.

A database data set (DBDS) group name and its associated DBDSs is defined in the DBRC RECON data set using the DBRC INIT.DBDSGRP command with parameters GRPNAME and MEMBERS.

- Adding resources to a group

  Use the CHANGE.DBDSGRP command with the GRPNAME and ADDDB parameters to add databases or areas to a database group.

  Use the CHANGE.DBDSGRP command with the GRPNAME and ADDMEM parameters to add DBDSs to a DBDS group.
- Deleting resources from a group

  Use the CHANGE.DBDSGRP command with the GRPNAME and DELDB parameters to delete databases or areas from a database group.

  Use the CHANGE.DBDSGRP command with the GRPNAME and DELMEM parameters to delete DBDSs from a DBDS group.
- Displaying resources in a group

  Use the LIST.DBDSGRP command or the online /RMLIST command to display the databases or areas in a database group, or the DBDSs in a DBDS group.

Groups defined for commands should contain only databases and areas for use with the DATAGROUP keyword.

Null Keywords

Null keywords are used to help clarify the syntax of the command being entered but have no effect on the command. Null keywords can be used within IMS commands at any time. However, they cannot be used with type-2 commands. Null keywords are reserved words. Do not use them to name system resources. The following are IMS null keywords:

- AFTER
- FOR
- ON
- SECURITY
- TO

Comments

You can add comments or notes to the end of a command. When you add comments, identify the end of the command by entering a period after the last parameter. If you do not enter a period, residual data from a 3270 affects command processing.

Passwords

Depending on the password security arrangement of the defined IMS system and the terminal where the command is entered, a password can be required after the command verb, or after a parameter or a keyword of certain commands.

If a command password is required, it must be entered immediately after /VERB and is normally enclosed in parentheses. No spaces or intervening characters are allowed between the /VERB and the left parenthesis.

When using MFS, if a command password is required, the installation-defined MFS format must contain a defined password field. The password must be entered into the password field defined by the format displayed. If command entry field of the format is defined to accept the command verb only, message DFS1051 is issued when an attempt is made to enter /VERB(password).

None of the IMS-supplied default formats (exception for the master terminal format) have a password field defined.

If a command is issued from a static terminal, you must use either the Security Maintenance utility, RACF, or an equivalent security product to require a command verb password. If a command is issued from a dynamic terminal, you must use RACF or an equivalent security product to require a command verb password associated with the user ID signed on to the dynamic terminal.

The /IAM, /LOCK, and /UNLOCK command keywords and parameters support passwords. Keyword and parameter passwords are defined through the security maintenance utility and only apply to static terminals. Keyword and parameter passwords entered with a command on a dynamic terminal are ignored. Keyword and parameter passwords entered with a command are also ignored when the security maintenance utility is not used or the resource is not defined with password security. This is also true for the /SET TRANSACTION and /SET CONVERSATION commands.

The use of passwords is not shown in the examples.

Command Processing in an IMSplex

In an IMSplex environment, IMS commands issued through OM can behave differently than when those same commands are issued to an individual IMS subsystem. IMSplex commands can be issued only through the OM API. Classic IMS commands can be issued through the OM API or to individual IMSs through end-user terminals, master terminals, system consoles, or AOI applications. The following sections describe some of the behavioral differences.

Command Routing

Commands that are issued to OM are, by default, routed to all the IMSplex components that are active and have registered interest in processing those

commands. If you want to route a command to one or more specific IMSs in the IMSplex, use the ROUTE() parameter on the command request.

OM selects one IMSplex member (that is, IMS or RM) that is registered for the command to be the command master for each command from the OM API. The command master performs global command actions where applicable. An XRF alternate system is not a command master candidate until it takes over.

IMSplex command responses may differ depending on which IMSplex member was the command master. For example, for a `QUERY TRAN NAME(tranname) QCNT (GT 1) SHOW(ALL)` command, only the command master returns the global queue counts, unless it does not have access to the Shared Queues (for example, the command master is local queues enabled).

Command Behavior and the Resource Manager

Depending on whether an IMSplex is defined with a Resource Manager (and there is a resource structure available to RM), command behavior can be affected. When a resource structure is not defined, resource status must be maintained on local IMSs in the IMSplex. In this case, commands have only a local effect.

If RM is defined with a resource structure in the IMSplex, RM maintains global resource information, including resource status. So, in this scenario, resource status is maintained both globally and locally. Usually, if a user signs off or a client shuts down, resources status is maintained globally but deleted locally.

If the DISABLE RESOURCE SHARING FOR STATIC ISC option is set, the command status for static ISC resources is always considered local as if there were no resource structure. Commands processed for a static ISC node or subpool only modify local status. Status is not updated in the resource structure. The purpose of the option is to remove the unique name restriction for static ISC-related resources so that static ISC LTERM and subpool names can be active multiple times concurrently in an IMSplex. For information about this option, see the Initialization exit routine, DFSINTX0, in the *IMS Version 9: Customization Guide*.

Another behavior that is worth noting is how command processing clients process classic commands (related to nodes, LTERMS, and users) that are routed to the entire IMSplex. In general, OM chooses one of the command processing clients in the IMSplex to be the “master” to coordinate the processing of the classic commands. Whether the master (or a non-master) client will process a classic command depends on where the command resource status is kept. If the command resource status is kept in a resource structure, the classic command will usually be processed by a non-master client where the command resource is active. If the command resource is not active on any of the command processing clients in the IMSplex, OM will still route the classic command to all clients in the IMSplex, but only the master client will process the command. If the classic command is being routed to all the clients in the IMSplex, command processing clients where the command resource is not active will reject the classic command.

IMSplex Command Format

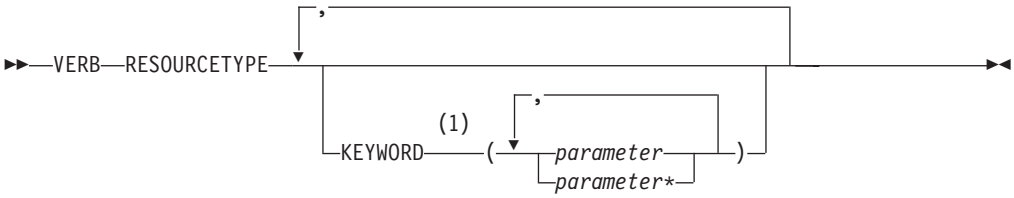
IMSplex commands use a different format from the format used by IMS classic commands. IMSplex commands can be issued only through the Operations Manager (OM) APIs by an automated operations program (AOP). An example of an AOP is the IBM-supplied TSO single point of control (SPOC) application. These commands cannot be entered from the system console, the MTO, an end-user

terminal, or an IMS AOI application program. IMSplex commands are not passed to the IMS AOI user exits and are also not logged to the IMS secondary master.

Almost all commands issued through the OM API (including the IMS classic commands) do not require a command recognition character (for example, a slash). If one is supplied, the OM API will ignore it. The only exception to this rule is the classic /EXIT command, which requires a slash when entered from a SPOC application.

Recommendation: IBM recommends against using command recognition characters for commands issued through the OM API (except for the /EXIT command).

The command format with common keywords is as follows:



Notes:

1 For some commands, KEYWORD is required, not optional.

The following is a list of the parameters with a short description of each.

VERB Identifies the action to be taken. The command verb can be abbreviated.

RESOURCETYPE
Identifies the type of resource the command acts upon. The following is a list of the resource types:

AREA An area resource in an IMS.

DATAGRP
The members of a data group in an IMS.

DB The database in an IMSplex.

IMSPLEX
A group of IMSs, CSL (SCI, OM, RM) members and CQSs.

LE Language Environment runtime parameter overrides.

MEMBER
An IMSplex component (For example, an IMS, OM, RM, or SCI).

OLC An online change process.

OLREORG
An online reorganization of a HALDB partition

STRUCTURE
A resource structure managed by RM.

TRANS
An IMS resource that is a message destination.

KEYWORD()
Optional command elements, depending upon the specific command.

parameter
Identifies an IMS-defined value or an installation-defined or created resource. The *parameter* can end with a wildcard (an asterisk - *). Parameters are listed in parentheses and must be replaced with parameter values. Multiple parameters are separated by a comma.

Command Characteristics

Command characteristics and operations (the way commands actually work) are affected by terminal security defaults, command recoverability, and whether commands are logged to the secondary master terminal, supported from an AOI application, mirrored on the XRF alternate, supported from the XRF alternate, and supported from LU 6.2 devices.

Terminal Security Defaults

The master terminal and remote terminal security defaults for IMS commands are described in Table 3.

Table 3. Terminal Security Defaults for IMS Commands

Master Terminal	Remote Terminal
/ACTIVATE	
/ALLOCATE	
/ASSIGN	
/BROADCAST	/BROADCAST
/CANCEL	/CANCEL
/CHANGE	
/CHECKPOINT	
/CLSDST	
/COMPT	
/CQCHKPT	
/CQQUERY	
/CQSET	
/DBDUMP	
/DBRECOVERY	
/DELETE	
/DEQUEUE	
2 /DIAGNOSE	/DIAGNOSE
/DISPLAY	
/END	/END
/ERESTART	
/EXCLUSIVE	/EXCLUSIVE
/EXIT	/EXIT
/FORMAT	/FORMAT
/HOLD	/HOLD
	/IAM
/IDLE	

Table 3. Terminal Security Defaults for IMS Commands (continued)

Master Terminal	Remote Terminal
/INITIATE	
/LOCK	/LOCK
/LOG	/LOG
/LOOPTEST	/LOOPTEST
/MODIFY	
/MONITOR	
/MSASSIGN	
/MSVERIFY	
/NRESTART	
/OPNDST	
/PSTOP	
/PURGE	
/QUIESCE	
/RCLSDST	/RCLSDST
	/RCOMPT
/RDISPLAY	/RDISPLAY
/RECOVER	
/RELEASE	/RELEASE
/RESET	/RESET
/RMxxxxxx	/RMLIST
/RSTART	
/RTAKEOVER	
/SECURE	
/SET	/SET
/SIGN	/SIGN
/SMCOPY	
/SSR	
/START	
/STOP	
/SWITCH	
/TERMINATE	
/TEST	/TEST
/TRACE	
/UNLOCK	/UNLOCK
/UPDATE	
/VUNLOAD	

Commands Recovered During Emergency Restart

Certain commands that successfully alter IMS resources are written to the system log as X'02' or X'22' log records and are reprocessed during emergency restart. IMS classic commands that are recovered during emergency restart write an X'02' log record. IMSplex commands that are recovered during emergency restart write an X'22' log record. Table 4 lists the commands, along with the exceptions, that are recovered during emergency restart.

Table 4. Commands Recovered During Emergency Restart

Command	Exceptions
/ASSIGN	
/CHANGE	/CHANGE DESCRIPTOR
/DELETE	
/END	
/EXCLUSIVE	
/EXIT	
/HOLD	
/LOCK	/LOCK LTERM, /LOCK NODE, /LOCK PTERM
/LOG	
/MONITOR	
/MSASSIGN	
/PSTOP	/PSTOP LTERM
/PURGE	/PURGE APPC, /PURGE LTERM
/RELEASE	
/RSTART	
/SET	/SET LTERM, /SET TRAN
/SMCOPY	
/START	/START APPC, /START ISOLOG, /START TRKAUTOARCH, /START PROG
/STOP	/STOP APPC
/TEST MFS	/TEST LINE, /TEST NODE, /TEST USER
/UNLOCK	/UNLOCK LTERM, /UNLOCK NODE, /UNLOCK PTERM, /UNLOCK SYSTEM
UPDATE TRAN	UPDATE TRAN START(TRACE), UPDATE TRAN STOP(TRACE)

Restriction: If an IMS outage (abend, modify, or cancel of IMS) occurs immediately after a command is entered, the command status might not be carried across an emergency restart or XRF takeover.

The command log records are logged asynchronously (no check write or wait write). If there is no other IMS activity that forces the log buffer to be written to the OLDS or WADS data set, the status set by the command, for restart purposes, did not occur.

There are many events in IMS where log records are check-written to the log. Any one of these events subsequent to the command causes the command log record to be written to the OLDS or WADS data set.

Commands Logged to the Secondary Master Terminal

The secondary master terminal provides a hardcopy log of system activity, including checkpoint information and error messages. If the hardcopy logging facility has been selected and not disabled (using the /SMCOPY command), certain IMS commands and responses are logged on the secondary master terminal.

You can select whether only commands issued from the master terminal are logged, only commands entered from other terminals are logged, or both are logged. Only the commands listed in Table 5 are logged to the secondary master terminal.

For more information on logging to the secondary master terminal, see *IMS Version 9: Operations Guide*.

Table 5. Commands Logged to the Secondary Master Terminal

Commands	Commands (continued)
/ACTIVATE	/PURGE
/ALLOCATE	/QUIESCE
/ASSIGN	/RCLSDST
/CHECKPOINT	/RCOMPT
/CLSDST	/RECOVER
/COMPT	/RMCHANGE
/CQCHKPT	/RMDELETE
/CQQUERY	/RMGENJCL
/CQSET	/RMINIT
/DBDUMP	/RMLIST
/DBRECOVERY	/RMNOTIFY
/DELETE	/RSTART
/DEQUEUE	/RTAKEOVER
2 /DIAGNOSE	/SECURE
/DISPLAY	/START
/IDLE	/STOP
3 /INITIATE OLREORG	/SWITCH
3 /MODIFY	/TERMINATE OLREORG
/MONITOR	/TRACE
3 /MSASSIGN	/UPDATE OLREORG
/OPNDST	/UNLOCK SYSTEM
/PSTOP	/VUNLOAD

Commands Supported from an AO Application

Automated operator (AO) applications are application programs that issue IMS operator commands using DL/I calls. AO applications can use two different DL/I calls to issue commands: CMD and ICMD. This section lists which IMS commands can be issued using each of these calls and describes command security for AO applications.

AO applications using the CMD call use the Security Maintenance utility for security. AO applications using the ICMD call use RACF (or the equivalent) or the Command Authorization user exit DFSCCMD0 for security.

For an overview of AO applications, see *IMS Version 9: Operations Guide*. For more detailed information about securing AO applications, see *IMS Version 9: Administration Guide: System*.

The Control Center, TSO SPOC, and REXX SPOC API are considered automated operator programs, which are different from AO applications. The automated operator programs do not issue commands through DL/I calls. For more information on these environments, see *IMS Version 9: Common Service Layer Guide and Reference*.

Using CMD Call

If you are using the CMD call to issue IMS commands, transactions can be defined as AO applications by the Security Maintenance utility or with AOI= on the TRANSACT system definition macro. AO transactions are invoked in the same way any IMS transaction is invoked. AO transactions run as IMS applications with the authority to issue a planned series of IMS commands. For example, an AO application can be called in by an AO transaction after a normal restart of IMS to start IMS resources. The AO application would consist of those commands regularly used by the master terminal operator (MTO) after IMS is active. Fast Path exclusive transactions cannot be defined as AO transactions.

When the Security Maintenance utility is used for Type 1 AOI security, the utility defines which transactions can issue commands, and which commands are allowed for each authorized transaction. However, when RACF (or an equivalent product) is used for Type 1 AOI security, the AOI= parameter on the TRANSACT macro defines which transactions can issue commands. RACF (or an equivalent product) defines which commands are allowed for each authorized transaction. These definitions in RACF give control to the authorized user who entered the transaction or to the transaction name itself. The AOI= parameter on the TRANSACT macro contains the controlling specification.

Table 6 shows the IMS commands that can be issued in an AO application using the CMD call. The table includes the IMS Command, the restrictions, and the application environment (MPP or BMP running under DB/DC or MPP or BMP running under DCCTL).

Table 6. Commands Supported from an AO Application Using CMD

		Application Environment	
IMS Command	**Restrictions**	**MPP or BMP Running Under DB/DC**	**MPP or BMP Running Under DCCTL**
/ACTIVATE		Yes	Yes
/ALLOCATE		Yes	Yes
/ASSIGN		Yes	Yes
/BROADCAST		Yes	Yes
/CHANGE		Yes	Yes
/CHECKPOINT	simple form (no keywords), SNAPQ, or STATISTICS	Yes	Yes
/CLSDST		Yes	Yes
/COMPT		Yes	Yes
/CQCHKPT		Yes	Yes

Table 6. Commands Supported from an AO Application Using CMD (continued)

			Application Environment	
	IMS Command	Restrictions	MPP or BMP Running Under DB/DC	MPP or BMP Running Under DCCTL
	/CQQUERY		Yes	Yes
	/CQSET		Yes	Yes
	/DBDUMP		Yes	Yes
	/DBRECOVERY		Yes	Yes
	/DELETE		Yes	Yes
	/DEQUEUE		Yes	Yes
2	/DIAGNOSE		Yes	Yes
	/DISPLAY		Yes	Yes
	/END	with keywords	Yes	Yes
	/EXCLUSIVE	with keywords	Yes	Yes
	/EXIT	with LINE and NODE keywords	Yes	Yes
	/FORMAT	with LTERM keyword	Yes	Yes
	/IDLE		Yes	Yes
3	/INITIATE	OLREORG	Yes	No
	/LOCK	without LTERM, NODE, or PTERM keywords	Yes	Yes
	/LOG		Yes	Yes
	/LOOPTEST		Yes	Yes
	/MONITOR		Yes	Yes
	/MSASSIGN		Yes	Yes
	/OPNDST		Yes	Yes
	/PSTOP		Yes	Yes
	/PURGE		Yes	Yes
	/QUIESCE		Yes	Yes
	/RDISPLAY		Yes	Yes
	/RECOVER		Yes	No
	/RMCHANGE		Yes	Yes
	/RMDELETE		Yes	Yes
	/RMGENJCL		Yes	Yes
	/RMINIT		Yes	Yes
	/RMLIST		Yes	Yes
	/RMNOTIFY		Yes	Yes
	/RSTART		Yes	Yes
	/SECURE		Yes	Yes
	/SMCOPY		Yes	Yes
	/SSR		Yes	Yes

Table 6. Commands Supported from an AO Application Using CMD (continued)

IMS Command	Restrictions	Application Environment: MPP or BMP Running Under DB/DC	Application Environment: MPP or BMP Running Under DCCTL
/START		Yes	Yes
/STOP		Yes	Yes
/SWITCH		Yes	Yes
/TERMINATE	OLREORG	Yes	No
/TEST MFS	with LINE/NODE keywords	Yes	Yes
/TRACE		Yes	Yes
/UNLOCK	without LTERM, NODE, PTERM, or SYSTEM keywords	Yes	Yes
/UPDATE	OLREORG	Yes	No
/VUNLOAD		Yes	No

If the 3270 message facility service (MFS) bypass option is selected, the output message specified is `MODNAME=DFSEDTN`, and the terminal is in preset mode, then the only command recognized by IMS is the `/RESET` command issued from an unformatted screen.All other commands are passed to the application program.

Using ICMD Call

Table 7 shows by application program type the commands that can be issued in an AO application using the ICMD call. This includes DRA thread running under DBCTL or DB/DC; BMP running under DBCTL; MPP, BMP, IFP running under DB/DC; and MPP, BMP, or IFP running under DCCTL. Commands that cause the IMS control region to terminate, such as `/CHE FREEZE`, cannot be issued.

You can secure the commands issued by an ICMD call using RACF (or the equivalent) or the Command Authorization user exit DFSCCMD0. RACF lets you specify which commands the applications can issue. DFSCCMD0 and RACF let you do authorization checking during ICMD processing. The authorization checking can be controlled either through the transaction name or the command name which is determined by the authorized user who entered the transactions. The AOI= parameter on the TRANSACT macro allows you to specify which authorization checking is to be used.

Table 7. Commands Supported from an AO Application Using ICMD

IMS Command	Application Environment: DBRA Thread Running Under DBCTL or DB/DC	Application Environment: BMP Running Under DBCTL	Application Environment: MPP, BMP, or IFP Running Under DB/DC	Application Environment: MPP, BMP, or IFP Running Under DCCTL
/ACTIVATE	N/A	N/A	Yes	Yes
/ALLOCATE	N/A	N/A	Yes	Yes
/ASSIGN	N/A	N/A	Yes	Yes
/BROADCAST	N/A	N/A	Yes	Yes

Table 7. Commands Supported from an AO Application Using ICMD (continued)

		Application Environment			
	IMS Command	DBRA Thread Running Under DBCTL or DB/DC	BMP Running Under DBCTL	MPP, BMP, or IFP Running Under DB/DC	MPP, BMP, or IFP Running Under DCCTL
	/CANCEL	N/A	N/A	No	No
	/CHANGE	Yes	Yes	Yes	Yes
	/CHECKPOINT (simple form)	Yes	Yes	Yes	Yes
	/CHECKPOINT SNAPQ	N/A	N/A	Yes	Yes
	/CHECKPOINT STATISTICS	Yes	Yes	Yes	Yes
	/CLSDST	N/A	N/A	Yes	Yes
	/COMPT	N/A	N/A	Yes	Yes
	/CQCHKPT	N/A	N/A	Yes	Yes
	/CQQUERY	N/A	N/A	Yes	Yes
	/CQSET	N/A	N/A	Yes	Yes
	/DBDUMP	Yes	Yes	Yes	N/A
	/DBRECOVERY	Yes	Yes	Yes	N/A
	/DELETE	Yes	Yes	Yes	Yes
	/DEQUEUE	Yes	Yes	Yes	Yes
2	/DIAGNOSE	Yes	Yes	Yes	Yes
	/DISPLAY	Yes	Yes	Yes	Yes
	/END	N/A	N/A	Yes	Yes
	/ERESTART	No	No	No	No
	/EXCLUSIVE	N/A	N/A	Yes	Yes
	/EXIT	N/A	N/A	Yes	Yes
	/FORMAT	N/A	N/A	Yes	Yes
	/HOLD	N/A	N/A	No	No
	/IAM	N/A	N/A	No	No
	/IDLE	N/A	N/A	Yes	Yes
3	/INITIATE	Yes	Yes	Yes	No
	/LOCK DATABASE	Yes	Yes	Yes	N/A
	/LOCK PROGRAM	Yes	Yes	Yes	Yes
	/LOCK TRANSACTION	N/A	N/A	Yes	Yes
	/LOG	Yes	Yes	Yes	Yes
	/LOOPTEST	N/A	N/A	Yes	Yes
	/MODIFY	No	No	No	No
	/MONITOR	N/A	N/A	Yes	Yes
	/MSASSIGN	N/A	N/A	Yes	Yes

Table 7. Commands Supported from an AO Application Using ICMD (continued)

	Application Environment			
IMS Command	**DBRA Thread Running Under DBCTL or DB/DC**	**BMP Running Under DBCTL**	**MPP, BMP, or IFP Running Under DB/DC**	**MPP, BMP, or IFP Running Under DCCTL**
/MSVERIFY	N/A	N/A	No	No
/NRESTART	No	No	No	No
/OPNDST	N/A	N/A	Yes	Yes
/PSTOP	Yes	Yes	Yes	Yes
/PURGE	N/A	N/A	Yes	Yes
/QUIESCE	N/A	N/A	Yes	Yes
/RCLSDST	N/A	N/A	No	No
/RCOMPT	N/A	N/A	No	No
/RDISPLAY	N/A	N/A	Yes	Yes
/RECOVER	Yes	Yes	Yes	No
/RELEASE	N/A	N/A	No	No
/RESET	N/A	N/A	No	No
/RMCHANGE	Yes	Yes	Yes	Yes
/RMDELETE	Yes	Yes	Yes	Yes
/RMGENJCL	Yes	Yes	Yes	Yes
/RMINIT	Yes	Yes	Yes	Yes
/RMLIST	Yes	Yes	Yes	Yes
/RMNOTIFY	Yes	Yes	Yes	Yes
/RSTART	N/A	N/A	Yes	Yes
/RTAKEOVER	No	No	No	No
/SECURE	N/A	N/A	Yes	Yes
/SET	N/A	N/A	No	No
/SIGN	N/A	N/A	No	No
/SMCOPY	N/A	N/A	Yes	Yes
/SSR	No	Yes	Yes	Yes
/START	Yes	Yes	Yes	Yes
/STOP	Yes	Yes	Yes	Yes
/SWITCH	Yes	Yes	Yes	Yes
/TERMINATE	Yes	Yes	Yes	No
/TEST MFS LINE or NODE or USER	N/A	N/A	Yes	Yes
/TRACE	Yes	Yes	Yes	Yes
/UNLOCK DATABASE	Yes	Yes	Yes	N/A
/UNLOCK PROGRAM	Yes	Yes	Yes	Yes

Table 7. Commands Supported from an AO Application Using ICMD (continued)

	Application Environment			
IMS Command	**DBRA Thread Running Under DBCTL or DB/DC**	**BMP Running Under DBCTL**	**MPP, BMP, or IFP Running Under DB/DC**	**MPP, BMP, or IFP Running Under DCCTL**
/UNLOCK TRANSACTION	N/A	N/A	Yes	Yes
/UPDATE	Yes	Yes	Yes	No
/VUNLOAD	Yes	Yes	Yes	N/A

The following commands and keywords are not permitted:

- `/CHECKPOINT` keywords ABDUMP, DUMPQ, FREEZE, PURGE, and QUIESCE
- The commands `/END`, `/EXIT`, and `/EXCLUSIVE` if they have no keywords
- `/LOCK` keywords LTERM, NODE, and PTERM
- `/UNLOCK` keywords LTERM, NODE, PTERM, and SYSTEM

Command Security When Using OM

For IMS command security, you have your choice of OM command security using RACF or equivalent, or IMS command security.

Recommendation: Use OM command security instead of IMS security.

By allowing OM to perform the security checks, commands which fail security authorization are not routed to IMS, reducing processing overhead and network traffic. When IMS command security is used, it is up to the user to ensure that all IMS systems use the same security profiles or user exits. If IMS systems in the same IMSplex use different security rules, the results of command security checking may be unpredictable. For more information about the IMS command verbs and keywords, resource names, and authorizations that are used for RACF security checking, see Appendix I, “IMS Commands, RACF Access Authorities and Resource Names Table,” on page 869.

Commands Mirrored on an XRF Alternate

Certain commands that successfully alter IMS resources are written to the system log as X'02' or X'22' log records. IMS classic commands that alter resources write an X'02' log record. IMSplex commands that alter resources write an X'22' log record. In an XRF environment, the X'02' or X'22' log records are read and processed by the alternate system as it tracks the active system. The effect of these commands is mirrored on the alternate system when it takes over and becomes the active system. These commands, along with exceptions, are listed in Table 8.

Table 8. List of IMS Commands Mirrored on an XRF Alternate

Command	Exceptions
/ASSIGN	/ASSIGN is mirrored only if the assignment does not involve the master terminal.
/CHANGE	
/DELETE	

Table 8. List of IMS Commands Mirrored on an XRF Alternate (continued)

Command	Exceptions
/END	
/EXCLUSIVE	
/EXIT	
/HOLD	
/LOCK	
/LOG	
/MONITOR	
/MSASSIGN	
/PSTOP	/PSTOP REGION
/PURGE	/PURGE APPC
/RELEASE	
/RESET	
/RSTART	
/SECURE	/SECURE APPC
/SET	
/SMCOPY	
/START	/START APPC, /START REGION
/STOP	/STOP APPC, /STOP REGION
/TEST	/TEST is usually not mirrored on an XRF alternate system. However, /TEST MFS is mirrored if the alternate system is tracking the state of the terminal that entered the /TEST MFS command.
/TRACE	/TRACE is usually not mirrored on an XRF alternate system. However, /TRACE SET LINE and /TRACE SET LINK are mirrored for lines and links that can be restarted. /TRACE SET NODE is mirrored for all node types.
/UNLOCK	/UNLOCK SYSTEM
UPDATE TRAN	UPDATE TRAN START(TRACE), UPDATE TRAN STOP(TRACE)

Commands Supported on the XRF Alternate

The commands that are valid on the XRF alternate, along with the exceptions, are shown in Table 9.

Table 9. List of Commands Supported on the XRF Alternate

Command	Exceptions
/ACTIVATE	
/CANCEL	
/CHANGE	
/CLSDST	
/COMPT	
/DISPLAY	/DISPLAY QCNT

Table 9. List of Commands Supported on the XRF Alternate (continued)

Command	Exceptions
/END	
/ERESTART	
/FORMAT	
/IDLE	
/NRESTART	
/OPNDST	
/PSTOP REGION	
QUERY AREA	
QUERY DB	
QUERY IMSPLEX	
QUERY MEMBER	
QUERY TRAN	
/RCLSDST	
/RCOMPT	
/RDISPLAY	
/RMLIST	
/SECURE	
/SMCOPY	
/START DC	
/START LUNAME	
/START REGION	
/START RTCODE	
/START SURVEILLANCE	
/STOP	
/STOP BACKUP	
/STOP DC	
/STOP LUNAME	
/STOP REGION	
/STOP RTCODE	
/STOP SURVEILLANCE	
/SWITCH	
/TEST	
/TRACE	
UPDATE TRAN SET(MAXRGN)	
UPDATE TRAN START|STOP(TRACE)	

Commands and Keywords Supported on an RSR Tracking Subsystem

Only commands and keywords required to manage RSR tracking are supported on an RSR tracking subsystem. For example, commands and keywords relating to transactions are not supported, because transactions are not allowed on a tracking subsystem. See Table 10 for the commands and keywords supported on an IMS DB/DC tracking subsystem. The subset of commands and keywords relating to databases are the only ones allowable on a DBCTL tracking subsystem.

Commands entered on an IMS tracking subsystem that are not supported on a tracking subsystem result in message

```
DFS136I  COMMAND xxxxxxxxxx INVALID FOR TRACKER
```

Commands entered on a DCCTL tracking subsystem that are not supported on the tracking subsystem or on DCCTL result in message

```
DFS136I COMMAND xxxxxxxxxx INVALID FOR DCCTL
```

Keywords entered on an IMS tracking subsystem that are not supported on a tracking subsystem result in message

```
DFS110  COMMAND KEYWORD yyyyyyyyyyyy INVALID FOR TRACKER
```

Keywords entered on a DBCTL or DCCTL tracking subsystem that are not supported on a tracking subsystem or with DBCTL or DCCTL result in message

```
DFS110  KEYWORD yyyyyyyyyyyy INVALID FOR DBCTL|DCCTL
```

Table 10 below lists the commands and keywords that are allowed on an IMS DB/DC RSR tracking. The table also includes comments.

Table 10. Commands and Keywords Allowable on an IMS DB/DC RSR Tracking Subsystem

Commands	Keywords	Comments
/ACTIVATE	NODE	
/ASSIGN	COMPONENT, ICOMPONENT, LINE, LTERM, NODE, PTERM, USER	
/BROADCAST	ACTIVE, LINE, LTERM, NODE, PTERM, USER	BTAM lines are not supported.
/CANCEL		
/CHECKPOINT	ABDUMP, DUMPQ, FREEZE, SNAPQ, STATISTICS	Simple checkpoint (that is, with no keywords) is also supported.
/CLSDST	NODE, USER	
/COMPT	CNS, CRD, NODE, NOTRDY, PCH, PDS, PRT, RDR, READY, TDS, UDS, USER, VID, WPM1, WPM2, WPM3	
/DBRECOVERY	AREA, DATABASE, DATAGROUP, LOCAL, NOFEOV	Command only valid for a DATABASE level tracking system. Keywords only valid for areas and databases tracked at the DATABASE level.
/DEQUEUE	LINE, LTERM, NODE, PTERM, PURGE, PURGE1, USER	
/DIAGNOSE	BLOCK, MODE, SNAP, TRAN	

Table 10. Commands and Keywords Allowable on an IMS DB/DC RSR Tracking Subsystem (continued)

Commands	Keywords	Comments
/DISPLAY	ACTIVE, AREA, ASSIGNMENT, DATABASE, DBD, DC, LINE, LTERM, MODE, MODIFY, MONITOR, NODE, OLDS, POOL, PTERM, SHUTDOWN STATUS, TRACE, TRACKING STATUS, USER, XTRC	• For `/DISPLAY POOL ALL`, only pools needed on the tracking subsystem are displayed. • For `/DISPLAY POOL pool`, some parameters are not supported. • For `/DISPLAY STATUS` with no keywords, only resources needed on a tracking subsystem are displayed. • For `/DISPLAY TRACE ALL`, only resources needed on a tracking subsystem are displayed.
/END	LINE, NODE, PTERM, USER	
/ERESTART	BUILDQ, CHECKPOINT, COLDBASE, COLDCOMM, COLDSYS, FORMAT, NOPASSWORD, NOTERMINAL, NOUSER, OVERRIDE, PASSWORD, TERMINAL, USER	
/FORMAT	LTERM	
/IDLE	LINE, NODE, NOSHUT	
/LOG		
/MODIFY	ABORT, COMMIT, LTERM, PASSWORD, PREPARE, TERMINAL	The following parameters are supported: ACBLIB, BLDL, FMTLIB, MODBLKS, and RACF.
/NRESTART	BUILDQ, CHECKPOINT, FORMAT, NOBUILDQ, NOPASSWORD, NOTERMINAL, NOUSER, PASSWORD, TERMINAL, USER	
/OPNDST	ID, LOGOND, MODE, NODE, Q, UDATA, USER, USERD	
/PSTOP	LINE, LTERM, PTERM	BTAM lines are not supported.
/PURGE	LINE, LTERM, PTERM	BTAM lines are not supported.
2 QUERY	AREA, DB, MEMBER	
/RCLSDST		
/RCOMPT	CNS, CRD, NODE, NOTRDY, PCH, PDS, PRT, RDR, READY, TDS, UDS, USER, VID, WPM1, WPM2, WPM3	
/RDISPLAY		
/RMxxxxxx	LTERM	Only applies to tracking subsystem RECON data set.
/RSTART	LINE, LOPEN, NODE, PTERM, USER	BTAM lines are not supported.
/RTAKEOVER	UNPLAN, NOREVERSE	
/SIGN	ON, USERD	
/SMCOPY	ON, TERMINAL	

Table 10. Commands and Keywords Allowable on an IMS DB/DC RSR Tracking Subsystem (continued)

Commands	Keywords	Comments
/START	AREA, AUTOARCH, DATABASE, DATAGROUP, DC, ISOLOG, LINE, LOCAL, LTERM, NODE, OLDS, PTERM, SERVGRP, USER, WADS	`/START AREA`, `/START DATABASE`, and `/START DATAGROUP` are only valid on a DATABASE level tracking subsystem. BTAM lines are not supported.
/STOP	ADS, AUTOARCH, DC, LINE, LTERM, NODE, OLDS, PTERM, SERVGRP, TRKAUTOARCH, USER, WADS	BTAM lines are not supported.
/SWITCH	CHECHPOINT, OLDS	
/TEST MFS	LINE, NODE, PTERM, USER	
/TRACE	LEVEL, LINE, MODULE,NODE, ON, OPTION, SET, TABLE, UNITYPE, USER, VOLUME	No LUMI, RETR, SCHD, or SUBS TABLE tracing is allowed on a tracking subsystem.
2 UPDATE AREA	START(ACCESS), STOP(ACCESS)	
2 UPDATE DATAGRP	START(ACCESS), STOP(ACCESS)	
2 UPDATE DB	START(ACCESS), STOP(ACCESS)	

BTAM Terminals in an RSR Environment

Since BTAM terminals are not supported for an RSR tracking subsystem, the following commands are not supported on an RSR tracking subsystem (that is, they will cause a `DFS058 COMMAND COMPLETED EXCEPT ...` message to be issued):

- `/BROADCAST LINE`
- `/BROADCAST PTERM`
- `/PSTOP LINE [PTERM]`
- `/PURGE LINE [PTERM]`
- `/RSTART LINE [PTERM]`
- `/START LINE [PTERM]`
- `/STOP LINE [PTERM]`

If the parameter ALL is used with one of these commands, the BTAM terminals affected are simply skipped.

Commands Supported from LU 6.2 Devices and OTMA

Only certain commands are valid from LU 6.2 devices and OTMA. The commands that are supported from LU 6.2 devices and OTMA are listed in Table 11.

Table 11. Commands Supported from LU 6.2 Devices and OTMA

Commands	Commands (continued)
/ALLOCATE*	/OPNDST*
/ASSIGN	/PSTOP
/BROADCAST*	/PURGE

Table 11. Commands Supported from LU 6.2 Devices and OTMA (continued)

Commands	Commands (continued)
/CHANGE*	/QUIESCE
/CHECKPOINT	/RDISPLAY
/CLSDST	/RMCHANGE*
/COMPT	/RMDELETE*
/DBDUMP	/RMGENJCL*
/DBRECOVERY	/RMINIT*
/DELETE	/RMLIST*
/DEQUEUE	/RMNOTIFY*
/DIAGNOSE	/RSTART
/DISPLAY	/SECURE
/ERESTART*	/SMCOPY
/IDLE	/SSR
/INITIATE OLREORG	/START
/LOG	/STOP
/MODIFY	/SWITCH
/MONITOR	/TERMINATE OLREORG
/MSASSIGN	/TRACE
/MSVERIFY	/UPDATE OLREORG
/NRESTART	

Note: * Multisegment commands are not supported by OTMA.

Terminal Security Defaults for LU 6.2 Devices and OTMA

Table 12 shows the terminal security defaults for IMS commands from LU 6.2 devices and OTMA when RACF is not used (RACF=NONE) and the command security exit (DFSCCMD0) is also not used. Only the commands shown in the table will be authorized in these environments.

Table 12. Terminal Security Defaults for IMS Commands from LU 6.2 Devices and OTMA

LU 6.2 Devices	OTMA
/BROADCAST	/LOC
/LOC	/LOG
/LOG	/RDISPLAY
/RDISPLAY	/UNL
/RMLIST	

Commands Supported by the OM API

Table 13 on page 40 is a list of IMS command verbs and primary keywords that can be issued through the Operations Manager (OM) API. The command verb can be specified in the long form or short form, as specified in column one and two respectively. The primary keyword is the first keyword following the command verb, as specified in column three.

Table 13. Commands Supported by the OM API

Command (long form)	Command (short form)	Primary Keywords
/ACTIVATE	/ACT	LINK, NODE
/ALLOCATE	/ALL	LU
/ASSIGN	/ASS	CLASS, CPRI, INPUT, LCT, LPRI, LTERM, NPRI, OUTPUT, PARLIM, PLCT, SEGNO, SEGSZ, TRAN, USER
/BROADCAST	/BRO	ACT, LINE, LTERM, MASTER, NODE, PTERM, USER
/CHANGE	/CHA	APPC, CCTL, CPLOG, DESC, DIR, FDR, LINK, NODE, PSWD, SUBSYS, SURV, TRAN, UOR, USER
/CHECKPOINT	/CHE	DUMPQ, FREEZE, PURGE, STATISTICS
/CLSDST	/CLS	NODE
/CQCHKPT	/CQC	SHAREDQ, SYSTEM
/CQQUERY	/CQQ	STATISTICS
/CQSET	/CQS	SHUTDOWN
/DBDUMP	/DBD	DB
/DBRECOVERY	/DBR	AREA, DB, DATAGRP
/DELETE	/DEL	DESC, PSWD, TERMINAL
DELETE	DEL	LE
/DEQUEUE	/DEQ	AOITKN, LINE, LTERM, LU, MSNAME, NODE, SUSPEND, TMEM, TRAN, USER
/DISPLAY	/DIS	ACT, AFFIN, AOITKN, APPC, AREA, ASMT, CCTL, CONV, CPLOG, CQS, DB, DBD, DESC, FDR, FPV, HSB, HSSP, LINE, LINK, LTERM, LU, MASTER, MODIFY, MSNAME, NODE, OASN, OLDS, OTMA, OVERFLOWQ, PGM, POOL, PSB, PTERM, Q, QCNT, RECOVERY, RTC, SHUTDOWN, STATUS, STRUC, SUBSYS, SYSID, TIMEOVER, TMEM, TRACE, TRACKING, TRAN, UOR, USER
/END	/END	LINE, NODE, USER
/ERESTART	/ERE	BACKUP, COLDBASE, COLDCOMM, COLDSYS, NULL
/EXCLUSIVE	/EXC	LINE, NODE, USER

Table 13. Commands Supported by the OM API (continued)

Command (long form)	Command (short form)	Primary Keywords
/EXIT	/EXI	CONV
/IDLE	/IDL	LINE, LINK, NODE
INITIATE	INIT	OLC, OLREORG
/LOCK	/LOC	DB, PGM, TRAN
/LOG	/LOG	
/MODIFY	/MOD	ABORT, COMMIT, PREPARE
/MONITOR	/MON	LINE
/MSASSIGN	/MSA	LINK, MSNAME, SYSID, TRAN
/NRESTART	/NRE	CHKPT, NULL
/OPNDST	/OPN	NODE
/PSTOP	/PST	LINE, LINK, LTERM, MSPLINK, REGION, TRAN
/PURGE	/PUR	APPC, FPPROG, FPRGN, LINE, LTERM, MSNAME, TRAN
QUERY	QRY	AREA, DB, IMSPLEX, LE, MEMBER, OLC, OLREORG STRUCTURE, TRAN
/QUIESCE	/QUI	NODE
/RDISPLAY	/RDI	MASTER
/RECOVER	/REC	ADD, REMOVE, START, STOP, TERMINATE
/RMCHANGE	/RMC	
/RMDELETE	/RMD	
/RMGENJCL	/RMG	
/RMINIT	/RMI	
/RMLIST	/RML	
/RMNOTIFY	/RMN	
/RSTART	/RST	LINE, LINK, MSPLINK, NODE, USER
/RTAKEOVER	/RTA	DUMPQ, FREEZE, UNPLAN
/SECURE	/SEC	APPC, OTMA
/SMCOPY	/SMC	MASTER, TERMINAL
/START	/STA	APPC, AREA, AUTOARCH, CLASS, DB, DATAGRP, DC, DESC, ISOLOG, LINE, LTERM, LU, MADSIOT, MSNAME, NODE, OLDS, OTMA, PGM, REGION, RTC, SB, SERVGRP, SLDSREAD, SUBSYS, SURV, THREAD, TMEM, TRAN, TRKARCH, USER, VGR, WADS, XRCTRACK

Table 13. Commands Supported by the OM API (continued)

Command (long form)	Command (short form)	Primary Keywords
/STOP	/STO	ADS, APPC, AREA, AUTOARCH, BACKUP, CLASS, DB, DATAGRP, DC, LINE, LTERM, LU, MADSIOT, MSNAME, NODE, OLDS, OTMA, PGM, REGION, RTC, SB, SERVGRP, SLDSREAD, SUBSYS, SURV, THREAD, TMEM, TRAN, USER, VGR, WADS, XRCTRACK
/SWITCH	/SWI	OLDS, SYSTEM, WADS
TERMINATE	TERM	OLC, OLREORG
/TEST	/TES	MFS
/TRACE	/TRA	SET
/UNLOCK	/UNL	DB, PGM, SYSTEM, TRAN
UPDATE	UPD	AREA, DATAGRP, DB, LE, OLREORG, TRAN
/VUNLOAD	/VUN	AREA

Chapter 2. Keyword Table and Definitions

All IMS keywords and their synonyms and environments, as well as keywords supported by the OM API, are listed in Table 14. The section following the table describes the keywords in greater detail.

A keyword listed in the OM keyword column indicates that it is the only accepted form supported by the OM API. If no keyword is listed in the OM keyword column, then the IMS keyword or its synonym is acceptable. If no synonyms are listed, none are permitted.

IMS ignores misspelled keywords; IMS issues an error message when it encounters an invalid keyword. In an IMSplex, invalid or misspelled keywords will result in an ″UNKNOWN POSITIONAL PARAMETER″ message.

`ALL=NO` or `ALL=DIS` can be specified on the KEYWD macro to prevent use of the `ALL` parameter. `ALL=NO` and `ALL=DIS` can be specified for all keywords. `ALL=NO` indicates that the `ALL` parameter is invalid on all the commands for the keyword. `ALL=DIS` indicates the `ALL` parameter is invalid on all the `/DISPLAY` commands for the keyword. Note, the KEYWD macro only applies to type-1 commands. For more information on the KEYWD macro, see *IMS Version 9: Customization Guide*.

Table 14. Keywords, Synonyms, and Their Environments

IMS Keyword	Synonym	OM Keyword	DB/DC	DBCTL	DCCTL
ABDUMP			X	X	X
ABORT			X	X	X
ACCESS			X	X	
ACTIVE	A, ACT	ACT	X	X	X
ADD			X	X	
ADS			X	X	
AFFINITY	AFFIN, AFF	AFFIN	X		X
ALLENTRIES	ALLENT		X	X	
AOITOKEN	AOITKN	AOITKN	X	X	X
APDB			X	X	
APMQ			X		X
APPC			X		X
AREA			X	X	
ASR			X		X
ASSIGNMENT	ASMT	ASMT	X		X
AUTO			X		X
AUTOARCH	ARCHIVE, AUTOARC	AUTOARCH	X	X	X
AUTOLOGON	AUTLGN		X		X
BACKUP	BU	BACKUP	X		X
BALGRP	BALG		X		X
BKERR			X	X	

© Copyright IBM Corp. 1974, 2003

Table 14. Keywords, Synonyms, and Their Environments (continued)

IMS Keyword	Synonym	OM Keyword	DB/DC	DBCTL	DCCTL
BUILDQ	BLDQ, BLDQS, BUILDQS		X		X
CAGROUP	CAGRP		X	X	
CANCEL			X	X	X
CCTL	CC	CCTL	X	X	
CHECKPOINT	CHECKPT, CHKPOINT, CHKPT	CHKPT	X	X	X
CLASS	CLS	CLASS	X		X
CMDAUTH			X		X
CMDAUTHE			X		X
CNS			X		X
COLDBASE	COLDB	COLDBASE	X	X	
COLDCOMM	COLDC	COLDCOMM	X		X
COLDSESS			X		X
COLDSYS	COLDS	COLDSYS	X	X	X
COMMIT			X	X	X
COMP			X	X	
COMPONENT	COMPT		X		X
CONTINUOUS	CONT		X		X
CONVERSATION	CONV	CONV	X		X
CPLOG			X		X
CPRI			X		X
CQS			X		X
CRD			X		X
DATABASE	DATABASES, DB, DBS	DB	X	X	
DATAGROUP	DATAGRP	DATAGRP	X	X	
DBALLOC			X	X	
DBD			X	X	
DBDS			X	X	
DBDSGRP			X	X	
DBS			X	X	X
DC			X		X
DESCRIPTOR	DESC, L62DESC	DESC	X		X
DIRECTORY	DIR	DIR	X		X
DONE			X		X
DUMPQ	DUMPQS	DUMPQ	X		X
EMHQ			X		X
ERRORABORT			X	X	
ERRORCONT			X	X	
EXIT			X		X

Table 14. Keywords, Synonyms, and Their Environments (continued)

IMS Keyword	Synonym	OM Keyword	DB/DC	DBCTL	DCCTL
FDR			X	X	
FORCE			X		X
FORCSESS	FORC		X		X
FORMAT	FMT		X	X	X
FPPROG			X		X
FPREGION	FPRGN	FPRGN	X		X
FPVIRTUAL	FPV	FPV	X	X	
FREEZE			X	X	X
GLOBAL			X	X	
GRSNAME	GRSN, GRS		X		X
HSB	HOTSTANDBY	HSB	X		X
HSSP			X	X	
ICOMPONENT	ICOMPT		X		X
ID			X		X
IMS			X	X	X
IMSPLEX			X	X	X
INDOUBT	IND		X	X	
INPUT			X		X
INTERVAL	INT, INTV		X	X	X
ISOLOG			X	X	X
JOBNAME	JBN, JOB		X	X	X
LE			X	X	X
LA			X		X
LEAVEGR			X		X
LEAVEPLEX			X	X	X
LEVEL			X		X
LINE	LINES	LINE	X		X
LINK			X		X
LMCT	LCT	LCT	X		X
LOCAL			X	X	
LOGOND	LGND		X		X
LOPEN			X		X
LPRI			X		X
LTERM	LTERMS	LTERM	X		X
LUNAME	LU, LUN	LU	X		X
MADSIOT			X	X	X
MASTER			X		
MAXRGN			X		X
		MEMBER			

Table 14. Keywords, Synonyms, and Their Environments (continued)

IMS Keyword	Synonym	OM Keyword	DB/DC	DBCTL	DCCTL
MODE	MODETABLE, MODETBL		X		X
MODIFY			X	X	X
MODULE			X		X
MONITOR	MON		X	X	X
MSDBLOAD			X		
MSG	MESSAGE		X		X
MSGAGE			X		X
MSNAME			X		X
MSPLINK			X		X
MULTSIGN			X		X
NOBACKOUT	NOBKO		X	X	
NOBMP			X	X	X
NOBUILDQ	NBLDQ, NOBLDQ		X		X
NOCHECK			X	X	
NOCMDAUTH			X		X
NOCMDAUTHE			X		X
NOCOMP			X	X	
NOCQSSHUT				X	
NODBALLOC			X	X	
NODE			X		X
NOFEOV			X	X	
NOOPEN			X	X	
NOPASSWORD	NPSWD		X	X	X
NOPFA			X	X	
NOREVERSE			X	X	X
NOSAVE			X		X
NOSHUT	NOS		X		X
NOTERMINAL	NOTER, NOTERM		X		X
NOTRANAUTH			X		X
NOTRANCMDS			X		X
NOTRDY			X		X
NOUSER			X		X
NPRI			X		X
OASN			X	X	X
OFFLINE			X	X	
OLC			X	X	X
OLDS			X	X	X
OLREORG		OLREORG			
OPEN			X	X	

Table 14. Keywords, Synonyms, and Their Environments (continued)

IMS Keyword	Synonym	OM Keyword	DB/DC	DBCTL	DCCTL
OPTION			X	X	X
OSAMGTF			X	X	
OTMA			X		X
OUTBND	OUTBOUND		X	X	X
OUTPUT			X		X
OVERFLOWQ	OFLWQ	OVERFLOWQ	X		X
OVERRIDE	OVER		X	X	X
PARLIM			X		X
PASSWORD	PASSWORDS, PSWD, PSWDS	PSWD	X	X	X
PCH			X		X
PDS			X	X	X
PI			X	X	
PITR			X	X	
PLMCT	PLCT	PLCT	X		X
POOL			X	X	X
PREPARE			X	X	X
PRIORITY	PRTY		X		X
PROGRAM	PGM, PGMS, PROG, PROGRAMS, PROGS	PGM	X	X	X
PRT			X		X
PRTKN	PRK		X	X	
PSB			X	X	X
PTERM	PTERMS	PTERM	X		X
PURGE			X	X	X
PURGE1	FIRST, SINGLE, SNGL		X		X
Q	QS, QUEUE, QUEUES	Q	X		X
QCNT			X		X
QMGR			X		X
QUIESCE			X		X
RCVTIME			X	X	
RCVTOKEN			X	X	
RDR			X		X
READNUM			X	X	
READY			X		X
RECOVERY	REC	RECOVERY	X	X	
RECOVGRP	RCVGRP		X	X	

Table 14. Keywords, Synonyms, and Their Environments (continued)

IMS Keyword	Synonym	OM Keyword	DB/DC	DBCTL	DCCTL
REGION	REGIONS, REG, REGS, MSGREG, MSGREGS, MSGREGION, MSREGIONS, THREAD	REGION	X	X	X
REMOTE			X		X
REMOVE	REM	REMOVE	X	X	
RESET			X	X	X
RTCODE	RTC	RTC	X		X
SAVE			X		X
SB			X	X	
SCHD			X	X	X
SEGNO			X		X
SEGSIZE	SEGSZ	SEGSZ	X		X
SERVGRP	SG	SERVGRP	X	X	X
SET			X	X	X
SHAREDQ	SHRQ	SHAREDQ	X		X
SHUTDOWN			X	X	X
SIDE			X		X
SNAP			X	X	X
SNAPQ			X		X
SNGLSIGN			X		X
SQTT			X		X
SSM	SUBSYSMEMBER		X	X	X
STAGLOBAL			X	X	
STALOCAL			X	X	
START	STA	START	X	X	
STATISTICS			X	X	X
STATUS			X	X	X
STOP	STO	STOP	X	X	
STRUCTURE	STRUC	STRUCTURE	X		X
SUBSYS	SUBSYSTEM, SUBSYSTEMS	SUBSYS	X	X	X
SURVEILLANCE	SUR, SURV, SURVEIL	SURV	X		X
SUSPEND			X		X
SYNCLEVEL	SYNCLV		X		X
SYNCSESS	SYNC		X		X
SYSID			X		X
SYSTEM			X		X
TABLE			X	X	X

Table 14. Keywords, Synonyms, and Their Environments (continued)

IMS Keyword	Synonym	OM Keyword	DB/DC	DBCTL	DCCTL
TAKEOVER	TKO, TKOVR		X		X
TCO			X		X
TDS			X		X
TERMINAL	TER, TERM, TERMINALS, TERMS, TERS	TERMINAL	X		X
THREAD			X	X	
TIMEOUT	TIMO		X		X
TIMEOVER			X		X
TMEMBER	TMEM	TMEM	X		X
TPIPE	TPI		X		X
TPNAME	TP, TPN		X		X
TRACE			X	X	X
TRACKING			X	X	X
TRACKING STATUS			X	X	X
TRANAUTH			X		X
TRANCMDS			X		X
TRANSACTION	TRANS, TRAN, TRANSACTIONS, TRANCODE, TRANCODES, TRS	TRAN	X		X
TRAP			X		X
TRKAUTOARCH	TRKARCH	TRKARCH	X	X	X
TRS			X	X	X
TYPE			X		X
UDATA			X		X
UDS			X		X
UNITYPE			X		X
UNPLAN			X	X	X
UOR			X	X	
USEDBDS	USEAREA		X	X	
USER	SUBPOOL	USER	X		X
USERD	USRD		X		X
USERID			X	X	X
VGRS	VGR	VGR	X		X
VID			X		X
VOLUME			X	X	X
VTAMPOOL	VPL		X		X
WADS			X	X	X
WPM1			X		X
WPM2			X		X

Table 14. Keywords, Synonyms, and Their Environments (continued)

IMS Keyword	Synonym	OM Keyword	DB/DC	DBCTL	DCCTL
WPM3			X		X
XRCTRACK			X	X	X
XTRC	XTRACE, EXTRACE		X	X	X

Notes:

[1] ISOLOG and TRKAUTOARCH are valid only for an RSR tracking subsystem.

ABDUMP

Requests abnormal termination and a dump of the following:

Application program executing in a region specified in the /STOP REGION command. The parameter stated with ABDUMP must be the name of a transaction currently scheduled for the specified region.

IMS control region when used with the /CHECKPOINT FREEZE, PURGE, or DUMPQ command.

Active control region in an XRF environment when used with either of the following commands:

- The /SWITCH command in the active system
- The /SWITCH command in the alternate system, when it is on the same processor as the active system

ABDUMP is also valid with /STOP BACKUP, when it is used to request abnormal termination and a dump of the backup system.

ABORT

Is used with the /MODIFY command to terminate an online change process. This keyword prevents the libraries from being switched and resets the QSTOP status for transactions. ABORT is also used with /CHANGE CCTL or /CHANGE UOR to back out changes for a unit of recovery.

ACCESS

Is used with the /START DATABASE command to indicate the type of access intended for the named database. This keyword overrides the database access specified at system definition.

ACTIVE

Refers to IMS resources currently in use with the following commands:

/BROADCAST, to refer to any terminal that:

- Is not stopped
- Is not locked (VTAM)
- Is not being polled (BTAM BSC)
- Has not had the /PSTOP command issued and completed against it

/DISPLAY, to refer to transaction codes, programs, regions, classes, terminals, the VTAM ACB, and subsystems. It is also used with /DISPLAY CCTL to show the active threads associated with the coordinator controller.

/SWITCH, to indicate that the command is being entered on the active system in an XRF environment.

ADS

Is used with the /STOP command to refer to an area data set.

AFFINITY
Is used with the /DISPLAY command to display VTAM affinities that the IMS subsystem has with specific nodes.

ALLENTRIES
Is used with the following commands:

/RECOVER REMOVE, to specify that the recovery list will be eliminated.

/RECOVER STOP, to specify that recovery will abort for all database data sets and areas (all entries) in the recovery list.

AOITOKEN
Is used with the following commands:

/DEQUEUE, to dequeue and discard messages associated with the specified AOI token name.

/DISPLAY, to show all the AOI tokens in the system.

/PSTOP REGION, to post an application program waiting for messages associated with an AOI token.

APDB
Is used with the /TRACE MONITOR command to monitor activity between application programs and databases, including Fast Path activity.

APMQ
Is used with the /TRACE MONITOR command to monitor activity between application programs and message queues, including Fast Path activity.

APPC
Is used with the following commands:

/DISPLAY, to show LU 6.2 related activities.

/PURGE, to purge transaction schedule requests from APPC/MVS.

/SECURE, to control the RACF security level from LU 6.2 application programs.

/START, to start transaction schedule requests from APPC/MVS.

/STOP, to stop transaction schedule requests from APPC/MVS.

AREA
Refers to the areas of a data entry database (DEDB). Areas are data sets that contain entire physical records. AREA is used with the following commands:

- /DBRECOVERY, to allow recovery by DEDB area.
- /DISPLAY, to show the DEDB areas being processed.
- QUERY , to display information about DEDB areas and area data set information in an IMSplex.
- /START, to make DEDB areas available for processing.
- /STOP, to stop the processing of DEDB areas.
- UPDATE, to change the status of area resources in an IMSplex.

ASR
Is used with the /CHANGE LINK and /CHANGE NODE commands to change the automatic session restart designation of a link or node.

ASSIGNMENT
Is used with the /DISPLAY command to refer to resource assignments.

AUTO
Is used with the /TRACE SET ON TIMEOUT command to support the IMS VTAM I/O Timeout facility. The AUTO keyword automatically issues a message along

with VTAM INACT and VTAM ACT commands to attempt activation of a node, if I/O exceeded the timeout value specified in the /TRACE command.

AUTOARCH
Is used with the /START and /STOP commands to start and stop automatic archiving of online log data sets.

AUTOLOGON
Is used with the following commands:

- /CHANGE, to specify that the autologon information previously specified by a user descriptor, output creation exit routine (DFSINSX0), or signon exit routine (DFSSGNX0) is being updated dynamically.
- /DISPLAY USER, to display the current autologon information for the specified user.

BACKUP
Is used with the following commands:

/ERESTART, to indicate that the control region is an alternate system in an XRF environment.

/STOP, to terminate an alternate system.

/SWITCH, to indicate the command is being entered on the alternate system.

BALGRP
Is used with the /DISPLAY QUEUE and /DISPLAY QCNT commands to display statistical information about the Fast Path balancing group queues.

BKERR
Is used with the /DISPLAY DATABASE command to display error queue elements associated with a DL/I or DEDB database. It also displays incomplete backouts preventing the use of a DL/I database.

BUILDQ
Is used with the /NRESTART and /ERESTART commands to request reconstruction of the message queue data sets.

CAGROUP
Is used with the following commands:

/RECOVER ADD, to specify that one or more change accumulation groups as defined in the RECON data sets will have their database data sets and areas added to the recovery list.

/RECOVER REMOVE, to specify that one or more change accumulation groups as defined in the RECON data sets will have their database data sets and areas removed from the recovery list.

/RECOVER STOP, to specify that recovery processing is to be stopped for the database data sets and areas in the specified change accumulation groups as defined in the RECON data sets.

CANCEL
Is used with the /STOP REGION command to cancel a message region, and it is only valid after a /STOP REGION ABDUMP command. It is also used with the /STOP APPC command to cause APPC/z/OS to initiate a shutdown sequence.

CCTL
Is used in /DISPLAY CCTL or /CHANGE CCTL to represent a CICS® subsystem that is connected to a DBCTL or IMS subsystem.

CHECKPOINT
Is used with the following commands:

/ERESTART, to identify the desired shutdown/restart sequence. Valid parameters are 0 or the checkpoint number provided by IMS when the system was shut down.

/NRESTART, to identify the desired shutdown/restart sequence. The only valid parameter for /NRESTART is 0.

/SWITCH OLDS, to take a simple checkpoint after switching OLDS.

CLASS

Is used with the /ASSIGN, /START, and /STOP commands to indicate that the specified command will affect the class-scheduling capability of IMS. /DISPLAY STATUS CLASS displays the status of classes. Valid parameters are one or more decimal numbers from 1 through 999 that represent class numbers assigned during IMS system definition.

This keyword is also used with /DISPLAY QUEUE, /PSTOP TRANSACTION, and /PURGE TRANSACTION.

CMDAUTH

Is used with the /ERESTART and /NRESTART commands to specify that both signon (user identification verification) and command authorization for static and ETO terminals are in effect at the end of the emergency restart.

CMDAUTHE

Is used with the /ERESTART and /NRESTART commands to specify that command authorization for ETO terminals (same as RCF=S on the startup parameter) is in effect at the end of the emergency restart. CMDAUTHE also resets command authorization for static terminals, if it was set.

CNS

Is used with the /COMPT and /RCOMPT commands to notify IMS that the console component is either operable or inoperable.

COLDBASE

Is used with the /ERESTART command to cold start the database component of IMS.

COLDCOMM

Is used with the /ERESTART command to indicate a data communication cold start.

COLDSESS

Is used with the /CHANGE LINK and /CHANGE NODE commands. COLDSESS is used with the /CHANGE LINK to reset the specified link or links. Use this keyword only after the /PSTOP command has been issued and completed against the link or links.

COLDSESS is used with the /CHANGE NODE command to set up the SLU P or FINANCE session so that the session can be cold started.

COLDSYS

Is used with /ERESTART to cold start both the database component and data communications components of IMS.

COMMIT

Is used to make the changes indicated by the /MODIFY PREPARE command or the INITIATE OLC PHASE(PREPARE) command. This keyword is also used with /CHANGE CCTL or /CHANGE UOR to commit changes for a unit of recovery.

COMP

Is used with the /TRACE SET PSB command to generate program communication block (PCB) and data compare statement images.

COMPONENT
Is used with the /ASSIGN command to refer to an output component member of a physical terminal complex. Valid parameters are numeric values from 1 through 4.

CONTINUOUS
Is used with the /RSTART LINK command to keep the link running by sending dummy data blocks when there is no data to be sent.

CONVERSATION
Is used only in IMS systems using conversational processing with the following commands:

/DISPLAY, to identify the status of conversations.

/EXIT, to terminate an active or held conversation; a valid parameter when terminating a held conversation is the identification of the conversation provided by IMS when the /HOLD command was used to suspend the conversation.

/RELEASE, to resume a previously held conversation; a valid parameter is the identification of the conversation provided by IMS when the /HOLD command was used to suspend the conversation.

/SET, to establish the destination of the next message entered on a terminal that is in a conversation.

CPLOG
Is used with the /DISPLAY command to display the value of the IMS execution parameter, CPLOG. The CPLOG keyword is also used with the /CHANGE command to change the value of the CPLOG IMS execution parameter. The CPLOG parameter specifies the number of log records written between system-generated checkpoints.

CPRI
Is used with the /ASSIGN command to refer to the current priority of a transaction code. The current priority of a transaction code is the current priority at which the transaction is scheduled. Valid parameters are 1- or 2-character numeric priorities, from 0 through 14.

CQS
Is used with the /DISPLAY command to display information about the IMS Common Queue Server subsystem that manages the IMS shared queues.

CRD
Is used with the /COMPT and /RCOMPT commands to notify IMS that the card component is operative or inoperative.

DATABASE
Refers to a database. Valid parameters are database names consisting of 1 to 8 alphanumeric characters. DATABASE is used with the following commands:

- /DBDUMP
- /DBRECOVERY
- /DELETE
- /DISPLAY
- /LOCK
- /START
- /STOP
- /UNLOCK

DATAGROUP

Refers to groups of DL/I databases and Fast Path DEDBs to be processed. Data groups are logical groupings of databases and areas; they allow simplified command processing for the databases and areas. DATAGROUP is used with the following commands:

- `/DBRECOVERY`, to allow recovery by data group.
- `/START`, to make data groups available for processing.
- `/STOP`, to stop the processing of data groups.

DATAGROUP is valid on active and RSR tracking subsystems.

DATAGRP

Refers to groups of DL/I databases and Fast Path DEDBs to be processed within an IMSplex. DATAGRP is used with the following commands:

- `QUERY`
- `UPDATE`

DB

Refers to a database to be processed within an IMSplex. DB is used with the following commands:

- `QUERY`
- `UPDATE`

DBALLOC

Is used with the `/START DATABASE` and `/START DATAGROUP` commands to indicate that databases will be allocated. If not specified, DBALLOC is the default for `/START DATABASE` 'dbname' commands only. For `/START DATABASE ALL` and `/START DATAGROUP` commands, the default is NODBALLOC. This keyword is not allowed on the RSR tracking system.

DBD

Is used with the `/DISPLAY` command, if Fast Path is installed, to display Fast Path and IMS databases, the PSBs that access them, and the type of access.

DBDS

Is used with the following commands:

`/RECOVER ADD`, to specify that one or more full function database data sets will be added to a recovery list.

`/RECOVER REMOVE`, to specify that one or more full function database data sets will be removed from the recovery list.

`/RECOVER STOP`, to specify that recovery processing is to be stopped for the specified full function database data sets.

DBDSGRP

Is used with the following commands:

`/RECOVER ADD`, to specify that one or more DBDS groups as defined in the RECON data sets will have their database data sets added to the recovery list.

`/RECOVER REMOVE`, to specify that one or more DBDS groups as defined in the RECON data sets will have their database data sets removed from the recovery list.

`/RECOVER STOP`, to specify that recovery processing will stop for the database data sets and areas making up the specified DBDS groups as defined in the RECON data sets.

DBS
Is used with the /DISPLAY MODIFY command to display databases or areas with work in progress that could prevent an online change from succeeding.

DC
Is used with the /START command to initiate IMS data communication processing, including processing VTAM logons. DC is used with the /STOP command to prohibit logging on to VTAM. DC is also used with /DISPLAY ACTIVE to show VTAM ACB status, as well as line and node activity.

DESCRIPTOR
Is used with the /CHANGE, /DELETE, and /DISPLAY commands to alter, delete, or show the LU 6.2 descriptor values. Valid parameters are from 1 to 8 alphanumeric characters long.

DIRECTORY
Is used with the /CHANGE command to restore the MFS Dynamic Directory back to its status after initialization.

DONE
Is used with the /IAM command to disconnect a terminal on a switched line from IMS.

DUMPQ
Is used with the /CHECKPOINT command to request that termination of IMS include DUMPQ functions.

DUMPQ is also used with the /RTAKEOVER command on an active subsystem for a planned RSR takeover. /RTAKEOVER DUMPQ indicates that a /CHECKPOINT DUMPQ type of shutdown is performed before the planned takeover occurs.

EMHQ
Causes global queue count information to display on the Expedited Message Handler queue (EMHQ) when both EMHQ and QCNT are specified. EMHQ is used with the following commands:

/DISPLAY LTERM, for terminal queues.

/DISPLAY NODE, for terminals connected to the node.

/DISPLAY USER, for users on terminal queues.

ERRORABORT
Is used with the /RECOVER START command and specifies that recovery will stop for all entries in the recovery list if any database data set or area is unable to complete recovery.

ERRORCONT
Is used with the /RECOVER START command and specifies that recovery is to continue if any database data set or area is able to complete recovery.

EXIT
Is used with the /DISPLAY TRACE and /TRACE EXIT commands and displays user exit tracing. Currently, only tracing for the DFSMSCE0 user exit is supported.

FDR
Is used with the /CHANGE and /DISPLAY commands to change or display information for IMS Fast Database Recovery regions.

FORCE
Is used with the /SWITCH command to cause immediate termination of the active system in an XRF environment.

FORCE is used with the /PSTOP command for VTAM links and is intended to be used when an MSC VTAM link will not cleanup and idle during normal PSTOP processing, even though VTAM has terminated the session.

FORCE also can be used in conjunction with some VTAM commands to idle and cleanup the VTAM link within IMS, so that a /RSTART LINK can be issued to restart the link again.

FORCE is also used with the /CLSDST command to re-initialize a hung node with I/O in progress.

FORCSESS
Is used with the /CHANGE command to allow session though message resynchronization was not successful. This change remains in effect until the next /CHANGE command or IMS cold start and overrides the FORCSESS/SYNCSESS specification on the TERMINAL or MSLINK macro. FORCSESS is valid for multiple system coupling (MSC) VTAM or Intersystem Communication (ISC).

FORMAT
Is used with /NRESTART and /ERESTART commands to request reformatting of one or more system data sets. Valid parameters are 2-character data set identifications.

FPPROG
Is used with the /PURGE command to terminate Fast Path message-driven programs by PSB name.

FPREGION
Is used with the /PURGE command to specify the region identifier of the message-driven programs to be terminated.

FPVIRTUAL
Is used with the /DISPLAY command to show DEDB areas loaded into z/OS data spaces for the Fast Path Virtual Storage Option (VSO). FPVIRTUAL is valid only on active DB/TM and DBCTL subsystems.

FREEZE
Is used with the /CHECKPOINT command to request termination of IMS.

FREEZE is also used with the /RTAKEOVER command on an active subsystem for a planned RSR takeover. /RTAKEOVER FREEZE indicates that a /CHECKPOINT FREEZE type of shutdown is performed before the planned takeover occurs.

GLOBAL
Is used with the /DBDUMP, /DBRECOVERY, /START, and /STOP commands to specify that the command applies to all subsystems sharing a database.

GRSNAME
Is used with the /START VGRS command to specify a VTAM generic resource name for an IMS subsystem when it joins a generic resource group.

HSB
Is used with the /DISPLAY command to show XRF related information.

HSSP
Is used with the /DISPLAY command to display high speed sequential processing (HSSP) information.

ICOMPONENT
Is used with the /ASSIGN command to refer to an input component number of a physical terminal complex. Valid parameters are numbers 1 through 4.

ID Is used with the /OPNDST NODE USER command to identify the other system half-session qualifier.

IMS PLEX
Is used with the IMSplex commands to refer to an IMSplex in a Parallel Sysplex® environment.

INDOUBT
Is used with the /DISPLAY CCTL command to show in-doubt threads.

INPUT
Is used with the following commands:

/ASSIGN, to change a logical terminal input assignment without changing the output assignment.

/DISPLAY LUNAME, to display information about LU 6.2 application programs.

/START LUNAME, to make the LUNAME available for inbound and synchronous outbound activities.

/STOP LUNAME, to stop the LUNAME for inbound and synchronous outbound activities.

/TRACE LUNAME, to trace the LUNAME for inbound and synchronous outbound activities.

INTERVAL
Is used with the /CHANGE SURVEILLANCE command to change the interval value of a particular surveillance mechanism in an XRF environment. The interval value controls how often the alternate system surveys the state of the active system.

ISOLOG
Is used with the /START command on an RSR tracking subsystem to request isolated log data (a gap in the log data) from the isolated log sender at the active site.

The ISOLOG keyword is valid only on an RSR tracking subsystem.

JOBNAME
Is used with the following commands to specify the job name on the JOB statement of a region's JCL member:

- /PSTOP REGION
- /START REGION
- /STOP REGION

LA
Is used with the /TRACE MONITOR command to monitor line and logical link events.

LE
Is used on DELETE, UPDATE, and QUERY commands to specify Language Environment® (LE) runtime parameter overrides.

LEAVEPLEX
In an IMSplex, this keyword is specified if the IMS that is being shut down is not going to rejoin the IMSplex. Specify the LEAVEPLEX keyword when you do not intend to bring the IMS back up in the IMSplex.

LEAVEGR
Is used with the /CHECKPOINT command to delete all affinities from the VTAM affinities table for an IMS subsystem during IMS shutdown. This keyword also causes IMS to leave the VTAM generic resources group.

LEVEL
Is used with the /TRACE command to indicate the extent of desired trace information. Valid parameters are numeric values from 1 through 4.

LINE
Refers to a communication line. Valid parameters are line numbers that are 1 to 4 characters long.

LINE is used with the following commands:

```
/ASSIGN           /LOOPTEST
/BROADCAST        /MONITOR
/DELETE PASSWORD  /PSTOP
/DEQUEUE          /PURGE
/DISPLAY          /RSTART
/END              /START
/EXCLUSIVE        /STOP
/EXIT             /TEST MFS
/IDLE             /TRACE
```

LINK
Refers to a logical link in a multiple systems configuration. Valid parameters are numbers 1 through 255.

LINK is used with the following commands:

```
/ACTIVATE         /MSASSIGN
/CHANGE           /PSTOP
/DISPLAY          /RSTART
/IDLE             /TRACE
```

LMCT
Is used with the /ASSIGN command to refer to the limit count of a transaction code. The limit count is the queue count number for which the current priority is changed to the limit priority. Valid parameters are numeric values from 1 through 65535.

LOCAL
Is used with the /DBDUMP, /DBRECOVERY, /MSASSIGN TRANSACTION, /START, and /STOP commands to specify that the command applies only to the subsystem in which the command is issued.

LOGOND
Is used with the /OPNDST command to indicate the logon descriptor used for session establishment for dynamic terminals.

LOPEN
Is used with the /RSTART LINE command to enable communication lines again when the next I/O operation occurs.

LPRI
Is used with the /ASSIGN command to refer to the limit priority of a transaction code. The LPRI is the priority to which this transaction is raised if the queue count is equal to, or exceeds, the limit count. Valid parameters are 1- or 2-character numeric priorities, from 0 through 14.

LTERM
Refers to a logical terminal. Valid parameters are logical terminal names consisting of 1 to 8 alphanumeric characters. The LTERM can be defined to IMS in the system definition or it can be dynamically introduced if ETO is enabled.

This keyword is used with the following commands:

```
/ASSIGN          /MODIFY
/BROADCAST       /PSTOP
/DELETE          /PURGE
/DEQUEUE         /RMxxxxxx
/DISPLAY         /SET
/FORMAT          /START
/IAM             /STOP
/LOCK            /UNLOCK
```

LTERM is also used with IMSplex commands.

LUNAME

Refers to the logical unit name of an LU 6.2 application program or destination. A network-qualified LU name is optional on the LUNAME keyword. A network-qualified LU name is 1 to 8 alphanumeric characters long (the network identifier of the originating system), followed by a period, and then followed by the LU name that is also 1 to 8 alphanumeric characters long. IMS commands that include network-qualified LU names must enclose the network-qualified LU name in single quotes; for example, 'NETID001.LUAPPC02'. No blanks are allowed in a network-qualified LU name.

An LU name that is not network-qualified consists of 1 to 8 alphanumeric characters.

This keyword can be used with the following commands:

```
/ALLOCATE           /START
/CHANGE DESCRIPTOR  /STOP
/DEQUEUE            /TRACE
/DISPLAY
```

MADSIOT

Is used with the following commands:

`/DISPLAY AREA`, to display only those areas that are currently in a long busy condition or are in long busy recovery mode.

`/START`, to specify the MADS I/O timing function. The `/START MADSIOT` command is valid only after the long busy handling function is disabled for a link failure, a structure failure, or a rebuild failure.

MASTER

Is used with the `/BROADCAST`, `/DISPLAY`, `/RDISPLAY`, and `/SMCOPY` commands to issue commands to or receive information from the IMS master terminal.

MAXRGN

Is used with the `/CHANGE TRANSACTION` command to change the maximum number of regions that can be simultaneously scheduled for a given transaction.

MEMBER

Is used with the `QUERY` command to refer to IMSplex components in an IMSplex.

MODE

Is used with the `/ALLOCATE`, `/CHANGE DESCRIPTOR`, `/DISPLAY`, `/OPNDST`, and `/RSTART LINK` commands to refer to the various modes in which VTAM terminals can operate. The mode parameter causes selection of the corresponding entry in the VTAM LOGON MODE table.

MODIFY

Is used with the `/DISPLAY` command to indicate the status of the resources to be deleted or changed.

MODULE

Is used with the `/TRACE` command to indicate the calls to trace. Valid parameters are DDM, MFS, and ALL.

MONITOR
Is used with the /TRACE command to activate the IMS Monitor. Valid parameters are LA, SCHD, APMQ, APDB, and ALL. MONITOR is also used with the /DISPLAY TRACE command.

MSDBLOAD
Is used with the /NRESTART command during warm starts to load MSDBs from the z/OS sequential data set MSDBINIT.

MSG
Is used with /TRACE SET ON TIMEOUT to support the IMS VTAM I/O Timeout facility. The MSG keyword is used with /TRACE SET ON TIMEOUT to issue a message when I/O has exceeded the timeout value specified in the /TRACE command.

MSGAGE
Is used with /DISPLAY QCNT command to display a count of messages in a shared queue in which the oldest message exceeds the specified message age (in days).

MSNAME
Is used with the following commands to refer to a logical link path in a multiple systems configuration:

- /BROADCAST MASTER
- /DEQUEUE
- /DISPLAY
- /MSASSIGN
- /MSVERIFY
- /PURGE
- /START
- /STOP

MSPLINK
Is used with the following commands to refer to a physical link in a multiple systems configuration:

- /DISPLAY ASSIGNMENT
- /MSASSIGN LINK
- /PSTOP
- /RSTART

MULTSIGN
Is used with the /ERESTART and /NRESTART commands to permit multiple signons for each user ID. It also indicates user identification verification is in effect at the end of emergency restart.

NOBACKOUT
Is used with the /START DATABASE command to suppress backout restart for a database not registered in DBRC.

NOBMP
Is used with the /ERESTART command to prevent backout of updates made by batch message processing programs active when the system failure occurred.

NOBUILDQ
Is used with the /NRESTART command to allow IMS to come up quickly when queues are reformatted.

NOCHECK
Is used with the /RECOVER START command. If a member of a recovery group is in the recovery list being started, NOCHECK specifies that the Online Recovery Service will not stop a timestamp recovery or a timestamp recovery to any prior point in time (PITR) if one of the following situations occurs:

- All members of the recovery group are not in the same recovery list.
- All members of the recovery group do not need to be recovered to the same point in time.

NOCMDAUTH
Is used with the /ERESTART and /NRESTART commands to reset command authorization on static and ETO terminals.

NOCMDAUTHE
Is used with the /ERESTART and /NRESTART commands to reset command authorization for static and ETO terminals. The command authorization is reset for static terminals because the command authorization for static terminals cannot exist without the command authorization for ETO terminals.

NOCOMP
Is used with the /TRACE SET PSB command to suppress PCB and data compare statement images.

NOCQSSHUT
Is used with the /CHE DUMPQ, /CHE FREEZE, or /CHE PURGE commands to **not** shut down the CQS address space when the IMS control region terminates. The CQS address space remains active and connected to the message queue structures. NOCQSSHUT is applicable only when IMS is running in a shared-queues environment. The default is to shut down the CQS address space when the IMS control region terminates.

NODBALLOC
Is used with the /START DATABASE and /START DATAGROUP commands to indicate that databases are not to be allocated. This is the default for /START DATABASE ALL and /START DATAGROUP commands. For /START DATABASE 'dbname' commands, the default is DBALLOC, and NODBALLOC must be specified if the database is not to be allocated. If not allocated, the database will be allocated when it is scheduled. This keyword is not allowed on the RSR tracking system.

NODBALLOC is valid only on an active subsystem.

NODE
Refers to a valid VTAM node. The node can be defined to IMS through system definition or dynamically introduced if ETO is enabled. If the NODE parameter is associated with ISC parallel sessions, the USER keyword should be used to specify the proper half-session. When the USER keyword is omitted, all half-sessions of the specified node are affected by the command.

When the NODE and USER keywords are both used, the USER keyword must follow the NODE keyword. For restrictions on the use of NODE USER keyword combinations, see Appendix E, "Commands with the NODE USER Keyword Combination," on page 851

NODE is used with the following commands:

```
/ACTIVATE           /EXIT
/ASSIGN             /IDLE
/BROADCAST          /LOCK
/CHANGE             /OPNDST
/CLSDST             /QUIESCE
/COMPT              /RSTART
/DELETE PASSWORD    /START
```

```
/DEQUEUE            /STOP
/DISPLAY            /TEST MFS
/END                /TRACE
/EXCLUSIVE          /UNLOCK
```

NODE is also used with IMSplex commands.

Several commands that affect NODEs are valid only for static terminals. These commands are valid only for static nodes:

- `/END NODE`
- `/EXCLUSIVE NODE`
- `/TEST MFS NODE`

The following commands are valid for static nodes with ISC parallel sessions:

- `/END NODE USER p1 USER p2`
- `/EXCLUSIVE NODE p1 USER p2`
- `/TEST MFS NODE p1 USER p2`

NOFEOV
Is used with the `/DBDUMP` and `/DBRECOVERY` commands to allow the commands to execute without forcing end-of-volume on the system log.

NOOPEN
Is used with the `/START DB`command to indicate that the named database should not be opened in the processing of this command.

NOPASSWORD
Is used with the `/NRESTART` command to negate the password security specifications of the IMS Security Maintenance utility.

NOPFA
Is used with the following commands to specify that DBRC is not to be notified of a change in status for a database or area:

- `/DBDUMP`
- `/DBRECOVERY`
- `/STOP DATABASE | AREA`

By using this keyword, DBRC does not prevent further authorizations for the database or area.

NOREVERSE
Is used with the `/RTAKEOVER UNPLAN` command to specify that an unplanned remote takeover cannot be reversed. The primary effect of this keyword is that the IMS tracking subsystem saves and processes all data it has received from an active site, regardless of whether that data was committed.

NOSAVE
Is used with the following commands:

`/ASSIGN`, to indicate that the assignment changes specified by the `/ASSIGN` command are lost when control blocks are deleted by the system when they are no longer needed.

`/CHANGE`, to indicate that the changed autologon information should not be retained.

NOSHUT
Is used with the `/IDLE LINE` command to stop input and output activities on

3270 remote BTAM lines without having to initiate a checkpoint shutdown. NOSHUT is not valid with the /IDLE NODE command.

NOTERMINAL
Is used with the /NRESTART command to negate the terminal security specifications of the IMS Security Maintenance utility.

NOTRANAUTH
Is used with the /ERESTART and /NRESTART commands to turn off transaction authorization. NOTRANAUTH is not the opposite of TRANAUTH. TRANAUTH sets transaction authorization and also turns on signon (user identification verification).

NOTRANCMDS
Is used with the /ERESTART and /NRESTART commands to negate transaction command security.

NOTRDY
Is used with the /COMPT and /RCOMPT commands to make a terminal component not ready.

NOUSER
Is used with the /NRESTART command to request that neither security authorization nor transaction authorization be in effect at the end of the normal restart.

NPRI
Is used with the /ASSIGN command to refer to the normal priority of a transaction code. The normal priority is assigned to the transaction as the current priority when the transaction queue count is less than the limit count. Valid parameters are 1- or 2-character numeric priorities from 0 through 14.

OASN
Is used with the /CHANGE and /DISPLAY commands to specify a specific outstanding recovery element related to an external subsystem being deleted from IMS. The origin application schedule number (OASN) is used by IMS to identify recovery information.

OFFLINE
Is used with the following commands:

/RECOVER ADD, to specify that the databases and areas undergoing recovery will be left offline after recovery is complete.

/RECOVER START, to specify that the database data sets and areas in the recovery list will remain offline after recovery is complete.

OLC
Is used with the INITIATE, QUERY, and TERMINATE commands to initiate, query, or terminate a global online change.

OLDS
Is used with the /DISPLAY, /START, /STOP, and /SWITCH commands to refer to online log data sets.

OLREORG
Is used with the INITIATE, /INITIATE, QUERY, /QUERY, TERMINATE, /TERMINATE, UPDATE, and /UPDATE commands to initiate, query, terminate, or update an online reorganization of a HALDB partition.

OPEN
Is used with the /START DBcommand to indicate that the named database should be opened in the processing of this command.

OPTION
Is used with the /TRACE command. When used with the PI keyword, OPTION indicates a request for a specific program isolation trace option. Valid parameters are LOG, TIME, and ALL. When used with the TABLE keyword, OPTION indicates whether or not the trace tables will be logged.

OSAMGTF
Is used with the /TRACE command to activate or deactivate tracing of GTF trace records for every OSAM buffer request of LOCATE or CREATE.

OSAMGTF keyword is also used with the /DISPLAY TRACE command to display whether the OSAM Buffer Trace facility is active or inactive.

OTMA
Is used with the /DISPLAY, /START, and /STOP commands to display, start, or stop IMS Open Transaction Manager Access (OTMA). OTMA is also used with the /SECURE command to control the RACF security level for OTMA.

OUTBND
Is used with the /CHANGEcommand to change the current outbound LU.

OUTPUT
Is used with the /ASSIGN command to change a logical terminal output assignment without changing the input assignment.

OUTPUT is also used with the following commands:

/DISPLAY LUNAME, to display LU 6.2 application programs.

/START LUNAME, to make the LU name available for asynchronous outbound activities.

/STOP LUNAME, to stop the LU name for asynchronous outbound activities.

/TRACE SET LUNAME, to trace the LU name for asynchronous outbound activities.

OVERFLOWQ
Is used with the /DISPLAY command to display queues on an overflow list structure in a shared-queues environment.

OVERRIDE
Is used with the /ERESTART command to restart the system after failure of power, machine, z/OS, or DBRC.

PARLIM
Is used with the /ASSIGN command when message regions are parallel processing a transaction. PARLIM is the maximum number of messages that can be enqueued, but not yet processed, by each active message region currently scheduled for this transaction. An additional message region is scheduled whenever the transaction queue count exceeds the PARLIM value multiplied by the number of regions currently scheduled for this transaction (for shared queues environments, the successful consecutive GU count is used instead of the queue count). Valid parameters are numeric values from 0 through 32767, and 65535, where 65535 disables transaction load balancing.

PASSWORD
Refers to the password security specifications of the IMS Security Maintenance utility. PASSWORD is used with the following commands:

/CHANGE, to replace a password known to the password security specifications of the IMS Security Maintenance utility with a new password.

/DELETE, to eliminate password security of the IMS Security Maintenance utility for a currently password-protected resource.

/MODIFY PREPARE, to add a password security definition.

/NRESTART, to request that the password security specifications of the IMS Security Maintenance utility be in effect at completion of the restart process.

PCH
Is used with the /COMPT and /RCOMPT commands to notify IMS that a punch component is either operable or inoperable.

PDS
Is used with the /COMPT and /RCOMPT commands to notify IMS that a print data set is operable or inoperable.

PI
Is used with the /TRACE command to request that program isolation be traced along with all calls to ENQ/DEQ and the DL/I buffer handler. PI is also used with the /DISPLAY TRACE command to display the status of the program isolation trace.

PITR
Is used with the /RECOVER START command and specifies that timestamp recovery (TSR) was specified or a point-in-time recovery is to be performed against the database data sets and areas specified with preceding /RECOVER ADD commands.

PLMCT
Is used with the /ASSIGN command to refer to the processing limit count of a transaction code. The processing limit count is the number of messages of this transaction code that a program can process in a single scheduling. Valid parameters are values from 0 through 65535.

POOL
Is used with the /DISPLAY command to refer to a buffer pool within the IMS system. Valid parameters are 3- and 4-character pool names.

PREPARE
Is used with the /MODIFY and INITIATE OLC commands to initiate the prepare phase to change resources online. These commands refer to local online change or global online change.

PRIORITY
Is used with the /DISPLAY Q command to request transaction scheduling information, such as queue count, class, and current priority.

PROGRAM
PROGRAM refers to an IMS application program. PROGRAM is used with the following commands:

```
/DELETE
/DISPLAY
/LOCK
/START
/STOP
/TRACE
/UNLOCK
```

Valid parameters are program names that are 1 to 8 alphanumeric characters long.

Use of the ALL parameter for /DISPLAY PROGRAM can be prevented by specifying ALL=NO on the KEYWD macro. For more information on the KEYWD macro, see *IMS Version 9: Customization Guide*.

PRT
Is used with the /COMPT and /RCOMPT commands to notify IMS that a printer component is either operable or inoperable.

PRTKN
Indicates the pseudo recovery token to be used with the /CHANGE CCTL command.

PSB
Is used with the /DISPLAY command if Fast Path is installed, to display Fast Path and IMS PSBs, the transactions they are processing, the databases they are accessing, and any Fast Path routing codes associated with the transactions.

This keyword is also used with the /TRACE command to record all DL/I calls issued for a specified PSB.

PTERM
Is used with the commands listed below to refer to a relative physical terminal:

```
/ASSIGN             /LOOPTEST LINE
/BROADCAST          /MONITOR LINE
/DELETE PASSWORD    /PSTOP LINE
/DEQUEUE            /PURGE LINE
/DISPLAY            /RSTART LINE
/END LINE           /START LINE
/EXCLUSIVE LINE     /STOP LINE
/EXIT LINE          /TEST MFS LINE
/IAM                /UNLOCK
/LOCK
```

Valid parameters are numbers corresponding to the relative position of the terminal on the line. When used without the LINE keyword on the /DISPLAY command, valid parameters are physical terminal attributes.

PURGE
Is used with the following commands:

CHECKPOINT, to request termination of IMS.

/DEQUEUE, to cancel all output messages enqueued on a logical terminal or destined to a physical terminal.

/PSTOP, to stop a channel-to-channel partner link that is not idle after a system failure.

PURGE1
Is used with the /DEQUEUE command to delete only the first message currently queued to an LTERM.

Q Is used with the /DISPLAY command to request message queue information. The keyword is also valid with /OPNDST NODE to request VTAM to queue SIMLOGON requests for VTAM/SNA-supported terminals.

QCNT
Is used with the following commands to display the global count of messages in a shared-queues environment:

- /DISPLAY
- /DISPLAY LINE
- /DISPLAY LINK
- /DISPLAY LTERM
- /DISPLAY LUNAME
- /DISPLAY MSNAME

- /DISPLAY NODE
- /DISPLAY Q
- /DISPLAY TMEMBER
- /DISPLAY TRANSACTION
- /DISPLAY USER

With this keyword, local queue counts are not displayed; any numbers displayed for local queue counts should be ignored.

QMGR
Is used with the /TRACE TABLEcommand to indicate that the queue manager trace is to be activated or deactivated.

QUIESCE
Is used with the /CHECKPOINT command to shut down the network only after normal processing is complete on all VTAM nodes.

RCVTIME
Is used with the /RECOVER REMOVE and /RECOVER START commands to specify a recovery to a point in time.

RCVTOKEN
Is used with the following commands:

- /DISPLAY RECOVERY, to specify the recovery token associated with the specific recovery list to display.
- /RECOVER ADD, to specify the unique identifier associated with the recovery list that the /RECOVER ADD command operates against.
- /RECOVER REMOVE, to specify the unique identifier associated with the recovery list that the /RECOVER ADD command operates against.
- /RECOVER START, to specify the logical terminal (including IMS master terminal and z/OS system console) that owns the recovery list that the /RECOVER START command operates against.

RDR
Is used with the /COMPT and /RCOMPT commands to notify IMS that a reader component is either operable or inoperable.

READNUM
Is used with the /RECOVER REMOVE and /RECOVER START commands to specify the number of input devices used in parallel during recovery.

READY
Is used with the /COMPT and /RCOMPT commands to make a terminal component ready.

RECOVERY
Is used in the /DISPLAY RECOVERY command to display the list of database data sets and areas being recovered by Online Recovery Service. Is also used with the /DISPLAY NODE RECOVERY or /DISPLAY USER RECOVERY commands to display status recovery settings for a node or user.

RECOVGRP
Is used with the /RECOVER ADD, /RECOVER REMOVE, and /RECOVER STOP commands to specify that the listed groups are recovery groups. A recovery group is a group of full-function databases or DEDB areas that are related.

REGION
Requests an action be taken with respect to a message processing region, batch message processing region, or Fast Path region.

REGION is used with the /ASSIGN, /DISPLAY ACTIVE, /PSTOP, /START, and STOP commands.

Valid REGION parameters used with /ASSIGN, /PSTOP, and /STOP are the decimal numbers from 1 through the number of regions defined during system definition.

When used with /PSTOP, REGION returns a QC status code to an application program active in the specified region.

Valid REGION parameters used with /START are names of IMS.JOBS members.

When used with /DISPLAY ACTIVE, REGION shows the status of active regions.

REMOTE

Is used with the /DISPLAY QCNT command to display remote LTERMs and remote transactions on shared queues.

REMOVE

Is used with the /RECOVER command to remove some or all database data sets and areas from the recovery list.

RESET

Is used with the /CHANGE command to indicate that an incomplete unit of work, identified by OASN and assigned to an external subsystem, will be deleted.

RTCODE

Refers to Fast Path routing codes and is used with the following commands:

/DISPLAY, to display the status of programs with specified routing codes.

/DISPLAY STATUS, to display routing codes that are not active.

/START, to activate routing codes and allow transactions associated with the routing codes to process.

/STOP, to prevent transactions associated with the routing code from being processed.

SAVE

Is used with the following commands:

/ASSIGN, to prevent deletion of user and LTERM control blocks across session and IMS restarts.

/CHANGE, to indicate that the changed autologon information should be retained.

/RECOVER STOP, to specify that the recovery list is not to be deleted when recovery is stopped.

SB

Is used with the /START and /STOP commands to dynamically enable and disable the usage of sequential buffering.

SCHD

Is used with the /TRACE TABLE command to indicate that the scheduler trace is to be activated or deactivated.

SEGNO

Is used with the /ASSIGN command to specify the maximum number of application program output segments allowed into the message queues per Get-Unique (GU). Valid parameters are numbers 0 through 65535. The default value is 0.

SEGSIZE

Is used with the /ASSIGN command to specify the maximum number of bytes

allowed for one output segment. Valid values are 0, 5, or greater than 5 to the maximum of 65535. The default value is 0.

SERVGRP
Is used with the /START or /STOP command to start or stop an RSR service group.

SERVGRP is valid on active and RSR tracking subsystems.

SET
Is used with the /TRACE command to indicate whether tracing of internal IMS events should be turned on or off. It is also used to start or stop the IMS Monitor.

SHAREDQ
Is used with the /CQCHKPT and /CQSET commands to specify the IMS shared queues.

SHUTDOWN
Is used with the /DISPLAY STATUS command to request the status of a shutdown-in-progress. Is also used with the /CQSET command to initiate a structure checkpoint when the IMS Common Queue Server shuts down.

SIDE
Is used with the /CHANGE DESCRIPTOR command to refer to the APPC/z/OS side information table entry name. The side information table entry contains default values for APPC conversation attributes such as LUNAME, TPNAME, or MODE.

Valid parameters are from 1 to 8 alphanumeric characters long.

SNAP
Is used with the /DIAGNOSE command and captures storage information and writes information to the OLDS as type X'6701' records.

SNAPQ
Is used with the /CHECKPOINT command to dump the contents of the message queues to the system log tape without causing a shutdown of IMS.

SNGLSIGN
Is used with the /ERESTART and /NRESTART commands to permit a single signon for each user ID. It also indicates that user ID verification is in effect at the end of an emergency restart.

SQTT
Is used with the /TRACE TABLE command to indicate that the shared queues trace is to be activated or deactivated.

SSM
Is used with the /START SUBSYSTEM command to allow external subsystem connection processing to be started if the SSM= parameter is not specified on the IMS startup JCL.

SSM can have one parameter associated with it. The subsystem member name can have a maximum of four characters.

STAGLOBAL
Is used with the /RECOVER ADD and /RECOVER START commands. If IRLM is available, STAGLOBAL specifies that a /START DB or AREA command will be processed on all IMSs that have the databases and areas defined when recovery successfully completes.

STALOCAL
Is used with the /RECOVER ADD and /RECOVER START commands to specify that the databases and areas that have entries in the recovery list will be brought online only on the IMS that ran the recovery. However, database data sets and areas with OFFLINE or STAGLOBAL specified on /RECOVER ADD commands will not be brought online. STALOCAL is ignored if PITR is specified.

START
Is used with the /RECOVER command to begin recovery for all database data sets and areas specified in the preceding /RECOVER ADD with the same recovery token that was specified in /RECOVER START.

STATISTICS
Is used with the /CHECKPOINT command to generate statistics records that are used for performance measurements and tuning an IMS system. Is also used with the /CQQUERY command to display statistics for coupling facility list structures.

STATUS
Is used with the /DISPLAY command to request the status of one or more system resources that are in exceptional states. STATUS also can be used with the SHUTDOWN keyword to request status of a shutdown-in-progress, and with the TRACKING keyword to get the status of the tracking system in an RSR complex.

STOP
Is used with the /RECOVER command to stop recovery for some or all database data sets and areas in the recovery list.

STRUCTURE
Is used with the following commands to refer to a primary coupling facility list structure used by IMS for shared queues or a coupling facility structure used by RM for resources:

- /CQCHKPT
- /CQQUERY
- /CQSET
- /DISPLAY
- QUERY

SUBSYS
Is used with the /CHANGE, /DISPLAY, /START, and /STOP commands to identify the subsystem name that the command action will affect.

SURVEILLANCE
Is used with the /CHANGE, /START, and /STOP commands to refer to the XRF surveillance function. During XRF surveillance, the active XRF IMS subsystem sends signals to the alternate system. From these signals, the alternate system learns that the active system is running satisfactorily. If these signals stop, then the alternate system knows that there might be a problem in the active system. The parameters used with the SURVEILLANCE keyword describe a particular surveillance mechanism:

LNK
ISC Link

RDS
Restart Data Set

LOG
System Log

SUSPEND
Is used with the /DEQUEUE command to reschedule all transactions that have been suspended.

SYNCLEVEL
Is used with the /CHANGE DESCRIPTOR command to refer to the APPC synchronous level. This level is either NONE or CONFIRM.

SYNCSESS
Is used with the /CHANGE command to force a message resynchronization when the session is initiated. This remains in effect until the next /CHANGE command or IMS cold start.

SYNCSESS, which overrides the FORCSESS/SYNCSESS specification on the TERMINAL macro or the MSPLINK macro, is valid for MSC VTAM links or ISC nodes.

SYSID
Is used with the /BROADCAST MASTER, /DISPLAY ASSIGNMENT, /MSASSIGN, and /MSVERIFY commands to refer to the system identification of a system in a multiple systems configuration.

SYSID is also used with the /DISPLAY TRANSACTION command to refer to the local or remote systems associated with the transaction. Valid values are from 1 through 255.

Each SYSID has an MSNAME. The MSNAME keyword can be used in place of the SYSID keyword.

SYSID TRANSACTION is used with the /DISPLAY command to refer to the IDs of the local and remote systems associated with the transaction.

SYSTEM
Is used with the /UNLOCK and /SWITCH commands to refer to an XRF capable IMS system. Is also used with the /CQCHKPT command to refer to an IMS Common Queue Server subsystem.

TABLE
Is used with the /TRACE command to start online tracing, which allows a given trace table to be started or stopped, and specifies which components should be traced in a common trace table. This keyword is also valid with /DISPLAY TRACE, which shows the status of table traces.

TAKEOVER
Is used with /TRACE LINK, /TRACE LINE, and /TRACE NODE to trace terminals during XRF takeover.

TCO
Is used with the /TRACE command to activate or deactivate tracing for the Time-Controlled Operation (TCO). TCO is also used with the /DISPLAY TRACE command to show whether the tracing for the TCO is active or inactive.

TDS
Is used with the /COMPT and /RCOMPT commands to notify IMS that a transmit data set is operable or inoperable.

TERMINAL
Refers to the terminal security specifications of the IMS Security Maintenance utility. TERMINAL is used with the following commands:

/DELETE, to eliminate terminal security for a specified transaction.

/MODIFY PREPARE, to add terminal security.

/NRESTART, to request that the terminal security specifications of the IMS Security Maintenance Utility be in effect at completion of the restart process.

/SMCOPY, to control the printing of command output to the secondary master terminal when commands are issued from terminals other than the master terminal.

THREAD
Is used with the /STOP command to stop a CCTL thread.

TIMEOUT
Is used with the /CHANGE SURVEILLANCE command to change a timeout value of a particular surveillance mechanism in an XRF environment. The timeout value controls the length of time the alternate system allows to elapse without a positive signal from the active system before requesting takeover or informing the operator of the potential failure.

TIMEOUT is also used with the /TRACE SET command to start or stop the IMS VTAM I/O Timeout facility. /DISPLAY TRACE TIMEOUT displays the status of the Timeout facility.

TIMEOVER
Supports the IMS VTAM I/O Timeout facility. The TIMEOVER keyword is used with the /DISPLAY command to display the VTAM nodes that have been waiting for a VTAM response for a longer time than your installation has specified with the /TRACE SET ON TIMEOUT command.

TMEMBER
Is used with the /DISPLAY command to display IMS Open Transaction Manager Access (OTMA) client status. It is also used with the /DISPLAY STATUS command to display all OTMA transaction pipes that are stopped.

It is used with the /DEQUEUE command to dequeue messages associated with an OTMA transaction pipe.

It is used with the /START and /STOP commands to cause IMS to resume or suspend sending of output to OTMA clients.

It is used with the /TRACE SET and /DISPLAY TRACE commands to set and display traces for OTMA transaction pipes.

TPIPE
Is used with the /DISPLAY command to display IMS Open Transaction Manager Access (OTMA) client status.

It is used with the /DEQUEUE command to dequeue messages associated with an OTMA transaction pipe.

It is used with the /START and /STOP commands to cause IMS to resume or suspend sending of output to OTMA clients.

It is used with the /TRACE command to trace transaction pipe activity for OTMA clients.

To remove TPIPEs from the IMS memory, perform a COLD START or COLD COMM.

TPNAME
Refers to the TPNAME of an LU 6.2 application program or destination. When used with the LUNAME keyword, TPNAME must follow LUNAME.

With any command other than /CHANGE DESCRIPTOR, a TPNAME parameter of DFSSIDE can be used to indicate that the specified LUNAME parameter represents the APPC/z/OS side information entry name.

The TPNAME parameter can be up to 64 alphanumeric characters long.

This keyword can be used with the following commands:

```
/ALLOCATE           /START
/CHANGE DESCRIPTOR  /STOP
/DEQUEUE            /TRACE
/DISPLAY
```

TRACE
Is used with the /DISPLAY command to examine the status and options of the current IMS traces.

TRACKING
Is used with the /DISPLAY command to display the RSR tracking status on either the active or RSR tracking subsystem.

TRACKING is valid on active or RSR tracking subsystems.

TRACKING STATUS
Is used with the /DISPLAY TRACKING STATUS command to display the status of a tracking subsystem.

TRANAUTH
Is used with the /ERESTART and /NRESTART commands to specify both the transaction authorization and user identification, with or without RACF, that will be in effect at the end of the restart.

TRANCMDS
Is used with the /ERESTART and /NRESTART commands to request that transaction command security be in effect at the end of the restart.

TRANCMDS is also used with the /MODIFY PREPARE command to add transaction command security.

TRANSACTION
Refers to a transaction code. Valid parameters are from 1 to 8 alphanumeric characters long.

This keyword is used with the following commands:

```
/ASSIGN          /PURGE
/CHANGE          QUERY
/DELETE          /SET
/DEQUEUE         /START
/DISPLAY         /STOP
/LOCK            /TRACE
/MSASSIGN        /UNLOCK
/PSTOP               UPDATE
```

TRAP
Is used with the /TRACE command to enable certain MFS serviceability enhancements. TRAP is also used with /DISPLAY TRACE to show the status of the trap trace.

TRKAUTOARCH
Is used to indicate that the RSR tracking system is to initiate a request to start the automatic archiving of the tracking log data sets. TRKAUTOARCH allows the user to start automatic archive after automatic archive has been terminated following archive data set full conditions.

TRS
Is used with the /DISPLAY MODIFY command to display transactions with work in progress that could prevent an online change from succeeding.

TYPE
Is used with the /CHANGE DESCRIPTOR command to refer to the APPC conversation type, which is either BASIC or MAPPED.

UDATA
Is used with the /OPNDST command to indicate the user data used with the logon. (The format of keyword and data: UDATA user data). The user data can include a logon descriptor name, a signon descriptor name, resource access control facility (RACF) access and authorization data, and user information used by a user exit.

UDS
Is used with the /COMPT and /RCOMPT commands to notify IMS that a user data set is operable or inoperable.

UNITYPE
Is used with the /TRACE command to trace all terminals of a specific type.

UNPLAN
Is used with the /RTAKEOVER command to specify an unplanned remote takeover. UNPLAN is valid only on the RSR tracking subsystem.

UOR
Is used with the /DISPLAY and /CHANGE commands to determine and specify units of recovery involved in protected conversations using RRS/z/OS.

USEDBDS
Is used with the /RECOVER ADD command and specifies that Online Recovery Service will not restore database data sets or areas specified in the command with image copies.

USER
A *user* can be a person signed onto a terminal or an ISC subpool. The USER keyword is used with:

- The /NRESTART command to request that user identification verification be in effect at the end of the normal restart.
- Commands that support ISC nodes, in which case the user can be either:
  - A dynamically created ISC user
  - An ISC static user, which is an ISC user defined by the SUBPOOL macro in a system definition.

  In this case, the USER keyword refers to a collection of logical terminals that can be allocated to an ISC node and used with parallel sessions. It is valid only for ISC nodes defined with a dynamically allocated set of logical terminals.

  When the USER keyword is specified with the NODE keyword, the USER keyword must follow the NODE specification. When the NODE is statically defined, the NODE USER combination can be used only with a command related to an ISC parallel session on that terminal.
- Commands that support dynamic terminals and users, in which case the user is created dynamically and can be ISC or non-ISC. If the term “dynamic user” is used in reference to signing on to a node, the user is a non-ISC user.

  In this case, the USER keyword refers to a collection of logical terminals associated with a user ID that can be signed on to an ACF/VTAM node. When specified with the NODE keyword, USER must follow the NODE specification. If the NODE USER descriptor is used to sign on, the USER keyword parameter must be the node name.

- The /DISPLAY command, in which case the user is signed onto a static terminal and is a non-ISC static user.

  The USER keyword refers to a user ID that is defined to an enhanced security product such as RACF. A user ID signed on to a statically defined terminal can be used only as a parameter with the /DISPLAY USER command.

Table 15 shows the commands that support the USER keyword. It also includes commands that support the USER keyword for dynamic users only.

Table 15. Commands That Support the USER Keyword

Commands	Commands for Dynamic Users Only
/ASSIGN	
/BROADCAST	/BROADCAST USER
/CHANGE	
/CLSDST	
/COMPT	
/DEQUEUE	
/DISPLAY	
/END	/END USER
/EXCLUSIVE	/EXCLUSIVE USER
/EXIT	/EXIT CONVERSATION USER
/NRESTART	
/OPNDST	
/QUIESCE	
/RSTART	/RSTART USER
/START	
/STOP	
/TEST	/TEST MFS USER
/TRACE	

USERD
Is used with the /OPNDST command to indicate the user descriptor used with the signon.

USERID
Is used in the DELETE LE, QUERY LE, and UPDATE LE commands. It is used as a filter to add, delete, or display entries in the Language Environment (LE) runtime parameter override table.

VGRS
Is used with the /START and /STOP commands to cause IMS to join or leave a VTAM generic resources group. For the /START command, this keyword also allows you to specify the generic resource name if IMS does not already have one.

VID
Is used with the /COMPT and /RCOMPT commands to notify IMS that a display component is either operable or inoperable.

VOLUME
Is used with the /TRACE SET ON|OFF TABLE|PI command to specify the volume of tracing to be done: low, medium, or high.

VTAMPOOL
Is used with the /ASSIGN USER command to deallocate an entire string of LTERMs defined for a user.

WADS
Is used with the /START, /STOP, and /SWITCH commands to add a write-ahead data set (WADS) to the pool of available WADS, to remove one from the pool, or to switch to a new one.

WPM1
Is used with the /COMPT and /RCOMPT commands to notify IMS that the word-processing component is either operable or inoperable.

WPM2
Is used with the /COMPT and /RCOMPT commands to notify IMS that the word-processing component is either operable or inoperable.

WPM3
Is used with the /COMPT and /RCOMPT commands to notify IMS that the word-processing component is either operable or inoperable.

XRCTRACK
Is used in the /START and /STOP commands. This keyword results in calls to the log router to initiate or terminate XRC tracking. It is only valid on a tracking IMS system.

XTRC
Is used with /DISPLAY TRACE to show the XTRC status in order to determine which external trace data set is active, whether the OLDS is used for trace logging, whether XTRC is usable, and the status of the XTRC data sets.

Part 2. IMS Commands

Chapter 3. /ACTIVATE 95
Format 95
Environments and Keywords 95
Usage 95
Examples 95
Example 1 for /ACTIVATE Command 95
Example 2 for /ACTIVATE Command 95

Chapter 4. /ALLOCATE 97
Format 97
Environments and Keywords 97
Usage 97

Chapter 5. /ASSIGN 99
Format 99
Environments and Keywords 100
Usage. 101
/ASSIGN in an IMSplex 102
Keywords and Parameters 103
Examples 110
Example 1 for /ASSIGN Command 110
Example 2 for /ASSIGN Command 110
Example 3 for /ASSIGN Command 111
Example 4 for /ASSIGN Command 111
Example 5 for /ASSIGN Command 111
Example 6 for /ASSIGN Command 111
Example 7 for /ASSIGN Command 112
Example 8 for /ASSIGN Command 112
Example 9 for /ASSIGN Command 112
Example 10 for /ASSIGN Command. 112
Example 11 for /ASSIGN Command. 112
Example 12 for /ASSIGN Command. 113
Example 13 for /ASSIGN Command. 113
Example 14 for /ASSIGN Command. 113
Example 15 for /ASSIGN Command. 113
Example 16 for /ASSIGN Command. 113
Example 17 for /ASSIGN Command. 114
Example 18 for /ASSIGN Command. 114
Example 19 for /ASSIGN Command. 114
Example 20 for /ASSIGN Command. 114

Chapter 6. /BROADCAST 115
Format 115
Environments and Keywords 115
Usage. 116
Examples 117
Example 1 for /BROADCAST Command 117
Example 2 for /BROADCAST Command 118
Example 3 for /BROADCAST Command 118
Example 4 for /BROADCAST Command 118
Example 5 for /BROADCAST Command 118
Example 6 for /BROADCAST Command 119
Example 7 for /BROADCAST Command 119

© Copyright IBM Corp. 1974, 2003

Chapter 7. /CANCEL 121
Format 121
Environments 121
Usage. 121
Example for /CANCEL Command 121

Chapter 8. /CHANGE 123
Format 123
Environments and Keywords 124
Usage. 125
Examples 133
Example 1 for /CHANGE Command 133
Example 2 for /CHANGE Command 133
Example 3 for /CHANGE Command 134
Example 4 for /CHANGE Command 134
Example 5 for /CHANGE Command 135
Example 6 for /CHANGE Command 135
Example 7 for /CHANGE Command 135
Example 8 for /CHANGE Command 136
Example 9 for /CHANGE Command 136
Example 10 for /CHANGE Command 136
Example 11 for /CHANGE Command 137
Example 12 for /CHANGE Command 137

Chapter 9. /CHECKPOINT 139
Format 139
Shutdown Checkpoint 139
Simple Checkpoint 139
Statistics Checkpoint 139
Environments and Keywords 139
Usage. 140
Examples 143
Example 1 for /CHECKPOINT Command. 143
Example 2 for /CHECKPOINT Command. 143
Example 3 for /CHECKPOINT Command. 143
Example 4 for /CHECKPOINT Command. 143
Example 5 for /CHECKPOINT Command. 144
Example 6 for /CHECKPOINT Command. 144

Chapter 10. /CLSDST. 145
Format 145
Environments and Keywords 145
Usage. 145
Example for /CLSDST Command. 147

Chapter 11. /COMPT 149
Format 149
Environments and Keywords 149
Usage. 150
Examples 150
Example 1 for /COMPT Command 150
Example 2 for /COMPT Command 150

Chapter 12. /CQCHKPT 153
Format 153
Environments and Keywords 153

Usage. 153
Examples 154
Example 1 for /CQCHKPT Command 154
Example 2 for /CQCHKPT Command 154

Chapter 13. /CQQUERY 155
Format 155
Environments and Keywords 155
Usage. 155
Examples 156
Example 1 for /CQQUERY Command 156
Example 2 for /CQQUERY Command 156

Chapter 14. /CQSET 157
Format 157
Environments and Keywords 157
Usage. 157
Example for /CQSET Command 158

Chapter 15. /DBDUMP 159
Format 159
Environments and Keywords 159
Usage. 159
Examples 161
Example 1 for /DBDUMP Command 161
Example 2 for /DBDUMP Command 162

Chapter 16. /DBRECOVERY 163
Format 163
Environments and Keywords 163
Usage. 163
Examples 167
Example 1 for /DBRECOVERY Command 167
Example 2 for /DBRECOVERY Command 167
Example 3 for /DBRECOVERY Command 167
Example 4 for /DBRECOVERY Command 168

Chapter 17. /DELETE 171
Format 171
Environments and Keywords 171
Usage. 172
Examples 172
Example 1 for /DELETE Command 172
Example 2 for /DELETE Command 172

Chapter 18. DELETE 175
Format 175
Environments and Keywords 175
Usage. 175
DELETE LE Output Fields 176
Return, Reason, and Completion Codes for DELETE LE 177
Examples 177
Example 1 for DELETE LE Command 177
Example 2 for DELETE LE Command 178

Chapter 19. /DEQUEUE 183

Format . . . 183
Environments and Keywords . . . 183
Usage. . . . 184
Examples . . . 188
Example 1 for /DEQUEUE Command . . . 188
Example 2 for /DEQUEUE Command . . . 188
Example 3 for /DEQUEUE Command . . . 188
Example 4 for /DEQUEUE Command . . . 188
Example 5 for /DEQUEUE Command . . . 188
Example 6 for /DEQUEUE Command . . . 189

Chapter 20. /DIAGNOSE . . . 191
Format . . . 191
Environments . . . 191
Usage. . . . 191
Example for /DIAGNOSE Command . . . 194

Chapter 21. /DISPLAY . . . 195
Format . . . 195
Environments and Keywords . . . 209
Usage. . . . 211
/DISPLAY ACTIVE . . . 211
/DISPLAY AFFINITY . . . 216
/DISPLAY AOITOKEN . . . 216
/DISPLAY APPC . . . 216
/DISPLAY AREA . . . 218
/DISPLAY ASSIGNMENT. . . . 220
/DISPLAY CCTL . . . 221
/DISPLAY CONVERSATION . . . 222
/DISPLAY CPLOG . . . 223
/DISPLAY CQS . . . 223
/DISPLAY DATABASE . . . 223
/DISPLAY DATABASE with BKERR . . . 225
/DISPLAY DBD . . . 226
/DISPLAY DESCRIPTOR. . . . 226
/DISPLAY FDR . . . 226
/DISPLAY FPVIRTUAL . . . 227
/DISPLAY HSB . . . 228
/DISPLAY HSSP . . . 229
/DISPLAY LINE . . . 230
/DISPLAY LINK . . . 230
/DISPLAY LTERM . . . 231
/DISPLAY LUNAME . . . 232
/DISPLAY LUNAME INPUT . . . 232
/DISPLAY LUNAME OUTPUT . . . 233
/DISPLAY LUNAME TPNAME . . . 233
/DISPLAY MASTER . . . 234
/DISPLAY MODIFY . . . 234
/DISPLAY MSNAME . . . 239
/DISPLAY NODE. . . . 239
/DISPLAY OASN SUBSYS . . . 242
/DISPLAY OLDS . . . 242
/DISPLAY OTMA. . . . 244
/DISPLAY OVERFLOWQ. . . . 245
/DISPLAY POOL . . . 246
Abbreviations Used in the /DISPLAY POOL Command . . . 249

/DISPLAY PROGRAM 254
/DISPLAY PSB 255
/DISPLAY PTERM 255
/DISPLAY Q 256
/DISPLAY QCNT 256
/DISPLAY RECOVERY 258
Recovery List Information 259
Recovery Progress Information 260
Recovery List Entry Information 260
/DISPLAY RTCODE 261
/DISPLAY SHUTDOWN STATUS 261
/DISPLAY STATUS (With No Keywords) 263
/DISPLAY STRUCTURE 265
/DISPLAY SUBSYS 265
Subsystem Status Terms 266
Dependent Region Status Terms 266
/DISPLAY SYSID TRANSACTION 267
/DISPLAY TIMEOVER 267
/DISPLAY TMEMBER 267
/DISPLAY TRACE 269
/DISPLAY TRACKING STATUS 270
/DISPLAY TRACKING STATUS on an Active Subsystem 271
/DISPLAY TRACKING STATUS on a Tracking Subsystem 273
/DISPLAY TRANSACTION 278
/DISPLAY UOR 279
/DISPLAY USER 280
Examples 282
Examples for /DISPLAY ACTIVE Command 282
Example for /DISPLAY AFFINITY Command 286
Example for /DISPLAY AOITOKEN Command 286
Example for /DISPLAY APPC Command 286
Examples for /DISPLAY AREA Command 287
Examples for /DISPLAY ASSIGNMENT Command 288
Examples for /DISPLAY CCTL Command. 293
Example for /DISPLAY CONVERSATION Command 294
Example for /DISPLAY CQS Command 296
Examples for /DISPLAY DATABASE Command 296
Example for /DISPLAY DBD Command 299
Example for /DISPLAY DESCRIPTOR Command 299
Example for /DISPLAY FDR Command 299
Example for /DISPLAY FPVIRTUAL Command. 300
Example for /DISPLAY HSB Command 301
Example for /DISPLAY HSSP Command 302
Example for /DISPLAY LINE Command 302
Examples for /DISPLAY LINK Command 303
Examples for /DISPLAY LTERM Command 304
Example for /DISPLAY LUNAME Command 305
Example for /DISPLAY MASTER Command 307
Examples for /DISPLAY MODIFY Command 307
Example for /DISPLAY MSNAME Command 310
Examples for /DISPLAY NODE Command 311
Example for /DISPLAY OASN Command 316
Example for /DISPLAY OLDS Command 316
Examples for /DISPLAY OTMA Command 316
Example for /DISPLAY OVERFLOWQ Command 316
Examples for /DISPLAY POOL Command 317

Example for /DISPLAY PROGRAM Command . . . 325
Example for /DISPLAY PSB Command . . . 326
Example for /DISPLAY PTERM Command . . . 327
Examples for /DISPLAY Q Command . . . 327
Example for /DISPLAY QCNT Command . . . 329
2 Examples for /DISPLAY RECOVERY Command . . . 330
Example for /DISPLAY RTCODE Command . . . 333
Examples of /DISPLAY SHUTDOWN STATUS Command . . . 333
Examples for /DISPLAY STATUS Command . . . 335
Example for /DISPLAY STRUCTURE Command . . . 338
Examples for /DISPLAY SUBSYS Command . . . 338
Example for /DISPLAY SYSID Command . . . 338
Example for /DISPLAY TIMEOVER Command . . . 339
Examples for /DISPLAY TMEMBER Command . . . 339
Examples for /DISPLAY TRACE Command . . . 339
Examples for /DISPLAY TRACKING STATUS Command . . . 344
Examples for /DISPLAY TRANSACTION Command . . . 346
Example for /DISPLAY UOR Command . . . 347
Examples for /DISPLAY USER Command . . . 348

Chapter 22. /END . . . 351
Format . . . 351
Environments and Keywords . . . 351
Usage . . . 351
Example for /END Command . . . 352

Chapter 23. /ERESTART . . . 353
Format . . . 353
Manual Restart of an XRF Alternate System . . . 353
Restart of IMS Following Loss of Virtual Storage Only . . . 353
Restart of IMS Following Loss of Virtual Storage and Message Queue Data Set Integrity . . . 353
Restart of IMS Following /ERESTART Failure of the Database Component 353
Restart of IMS Following /ERESTART Failure of Communication Component 353
Restart of IMS Following /ERESTART Failure of Both the Database and Communication Components . . . 354
Environments and Keywords . . . 354
Usage . . . 355
Examples . . . 361
Example 1 for /ERESTART Command . . . 361
Example 2 for /ERESTART Command . . . 361
Example 3 for /ERESTART Command . . . 362
Example 4 for /ERESTART Command . . . 362
Example 5 for /ERESTART Command . . . 362

Chapter 24. /EXCLUSIVE . . . 365
Format . . . 365
Environments and Keywords . . . 365
Usage . . . 365
Examples . . . 366
Example 1 for /EXCLUSIVE Command . . . 366
Example 2 for /EXCLUSIVE Command . . . 366

Chapter 25. /EXIT . . . 367
Format . . . 367
Environments and Keywords . . . 367

Usage. 367
Examples 368
Example 1 for /EXIT Command 368
Example 2 for /EXIT Command 368
Example 3 for /EXIT Command 368
Example 4 for /EXIT Command 369
Example 5 for /EXIT Command 369

Chapter 26. /FORMAT 371
Format 371
Environments and Keywords 371
Usage. 371
Example for /FORMAT Command 372

Chapter 27. /HOLD. 373
Format 373
Environments 373
Usage. 373
Example for /HOLD Command 373

Chapter 28. /IAM 375
Format 375
Environments and Keywords 375
Usage. 375
Examples 376
Example 1 for /IAM Command. 376
Example 2 for /IAM Command. 376
Example 3 for /IAM Command. 376

Chapter 29. /IDLE 377
Format 377
Environments and Keywords 377
Usage. 377
Examples 379
Example 1 for /IDLE Command 379
Example 2 for /IDLE Command 379
Example 3 for /IDLE Command 379
Example 4 for /IDLE Command 379

Chapter 30. /LOCK. 381
Format 381
Environments and Keywords 381
Usage. 381
Examples 383
Example 1 for /LOCK Command 383
Example 2 for /LOCK Command 383
Example 3 for /LOCK Command 383
Example 4 for /LOCK Command 384
Example 5 for /LOCK Command 384

Chapter 31. INITIATE 385
Format 385
Environments and Keywords 385
INITIATE OLC. 386
INITIATE OLC Output Fields 392
INITIATE Return and Reason Codes 394

INITIATE Completion Codes . . . 397
Examples for INITIATE OLC Command . . . 409

Chapter 32. /LOG . . . 413
Format . . . 413
Environments . . . 413
Usage. . . . 413
Example for /LOG Command . . . 413

Chapter 33. /LOOPTEST . . . 415
Format . . . 415
Environments and Keywords . . . 415
Usage. . . . 415
Example for /LOOPTEST Command . . . 415

Chapter 34. /MODIFY . . . 417
Format . . . 417
Environments and Keywords . . . 417
Usage. . . . 417
Examples . . . 421
Example 1 for /MODIFY Command . . . 421
Example 2 for /MODIFY Command . . . 422
Example 3 for /MODIFY Command . . . 422

Chapter 35. /MONITOR . . . 425
Format . . . 425
Environments and Keywords . . . 425
Usage. . . . 425
Example for /MONITOR Command . . . 425

Chapter 36. /MSASSIGN . . . 427
Format . . . 427
Environments and Keywords . . . 427
Usage. . . . 427
Examples . . . 428
Example 1 for /MSASSIGN Command . . . 428
Example 2 for /MSASSIGN Command . . . 429
Example 3 for /MSASSIGN Command . . . 429
Example 4 for /MSASSIGN Command . . . 429

Chapter 37. /MSVERIFY . . . 431
Format . . . 431
Environments and Keywords . . . 431
Usage. . . . 431
Examples . . . 431
Example 1 for /MSVERIFY Command . . . 431
Example 2 for /MSVERIFY Command . . . 432

Chapter 38. /NRESTART . . . 435
Format . . . 435
Environments and Keywords . . . 437
Usage. . . . 437
Examples . . . 442
Example 1 for /NRESTART Command . . . 442
Example 2 for /NRESTART Command . . . 442
Example 3 for /NRESTART Command . . . 443

Example 4 for /NRESTART Command 443
Example 5 for /NRESTART Command 443
Example 6 for /NRESTART Command 443
Example 7 for /NRESTART Command 444
Example 8 for /NRESTART Command 444
Example 9 for /NRESTART Command 444

Chapter 39. /OPNDST 447
Format 447
Environments and Keywords 447
Usage. 448
Examples 450
Example 1 for /OPNDST Command 450
Example 2 for /OPNDST Command 451
Example 3 for /OPNDST Command 451
Example 4 for /OPNDST Command 451
Example 5 for /OPNDST Command 451

Chapter 40. /PSTOP 453
Format 453
Environments and Keywords 453
Usage. 454
Examples 456
Example 1 for /PSTOP Command 456
Example 2 for /PSTOP Command 457
Example 3 for /PSTOP Command 457
Example 4 for /PSTOP Command 457
Example 5 for /PSTOP Command 458
Example 6 for /PSTOP Command 458
Example 7 for /PSTOP Command 458
Example 8 for /PSTOP Command 458
Example 9 for /PSTOP Command 458
Example 10 for /PSTOP Command 459
Example 11 for /PSTOP Command 459
Example 12 for /PSTOP Command 459
Example 13 for /PSTOP Command 459
Example 14 for /PSTOP Command 460

Chapter 41. /PURGE 461
Format 461
Environments and Keywords 461
Usage. 462
Examples 463
Example 1 for /PURGE Command 463
Example 2 for /PURGE Command 463
Example 3 for /PURGE Command 463
Example 4 for /PURGE Command 464
Example 5 for /PURGE Command 464
Example 6 for /PURGE Command 464
Example 7 for /PURGE Command 464

Chapter 42. QUERY 465
Format 465
Environments and Keywords 468
QUERY AREA 470
QUERY AREA Output Fields 471

QUERY AREA STATUS 473
Return, Reason, and Completion Codes for the QUERY AREA Command 474
QUERY AREA Compared to Other Commands 475
Examples for QUERY AREA Command 475
QUERY DB. 478
QUERY DB Output Fields 479
QUERY DB Status 480
Return, Reason, and Completion Codes for the QUERY DB Command 481
QUERY DB Compared to Other Commands. 482
Examples for QUERY DB Command 482
QUERY IMSPLEX 487
QUERY IMSPLEX Output Fields 489
QUERY IMSPLEX Status 489
QUERY IMSPLEX Types. 490
QUERY IMSPLEX Subtypes 490
Return, Reason, and Completion Codes for QUERY IMSPLEX. 492
Examples for QUERY IMSPLEX Command 493
QUERY LE 495
QUERY LE Output Fields 496
Return, Reason, and Completion Codes for QUERY LE 497
Examples for QUERY LE Command 497
QUERY MEMBER 501
QUERY MEMBER Output Fields 502
QUERY MEMBER Attributes 503
QUERY MEMBER Status 503
Return, Reason, and Completion Codes for QUERY MEMBER. 506
Examples for QUERY MEMBER Command 506
QUERY OLC 509
QUERY OLC Output Fields 510
Return, Reason, and Completion Codes for QUERY OLC. 512
Examples for QUERY OLC Command 514
QUERY STRUCTURE. 516
QUERY STRUCTURE Output Fields 516
Return, Reason, and Completion Codes for QUERY STRUCTURE 518
Example for QUERY STRUCTURE 519
QUERY TRAN 520
QUERY TRAN Compared to Other Commands 524
QUERY TRAN Output Fields 524
Return, Reason, and Completion Codes for QUERY TRAN 527
Examples for QUERY TRAN Command 529

Chapter 43. /QUIESCE 537
Format 537
Environments and Keywords 537
Usage. 537
Example for /QUIESCE Command 538

Chapter 44. /RCLSDST 539
Format 539
Environments 539
Usage. 539
Example for /RCLSDST Command 540

Chapter 45. /RCOMPT 541
Format 541
Environments and Keywords 541

Usage. 541
Example for /RCOMPT Command 542

Chapter 46. /RDISPLAY 543
Format 543
Environments 543
Usage. 543
Examples 543
Example 1 for /RDISPLAY Command 543
Example 2 for /RDISPLAY Command 543

Chapter 47. /RECOVER 545
Format 545
Environments and Keywords 546
Usage. 547
/RECOVER ADD. 547
/RECOVER REMOVE 550
/RECOVER START. 552
/RECOVER STOP 554
/RECOVER TERMINATE. 555
Examples 556
Examples for /RECOVER ADD Commands 556
Examples for /RECOVER REMOVE Command 557
Examples for /RECOVER START Command 558
Examples for /RECOVER STOP Command 560
Example for /RECOVER TERMINATE Command 560

Chapter 48. /RELEASE 561
Format 561
Environments and Keywords 561
Usage. 561
Example for /RELEASE Command 561

Chapter 49. /RESET 563
Format 563
Environments 563
Usage. 563
Example for /RESET Command 563

Chapter 50. /RMxxxxxx 565
Format 565
Environments and Keywords 565
Usage. 565
Examples 567
Example for /RMCHANGE Command 567
Example for /RMDELETE Command 568
Example for /RMGENJCL Command 568
Example for /RMINIT Command 568
Example for /RMLIST Command 569
Example for /RMNOTIFY Command 571

Chapter 51. /RSTART. 573
Format 573
Environments and Keywords 573
Usage. 574
Examples 576

Example 1 for /RSTART Command 576
Example 2 for /RSTART Command 576
Example 3 for /RSTART Command 576
Example 4 for /RSTART Command 576
Example 5 for /RSTART Command 577
Example 6 for /RSTART Command 577
Example 7 for /RSTART Command 577
Example 8 for /RSTART Command 577

Chapter 52. /RTAKEOVER 579
Format 579
/RTAKEOVER for an Active Subsystem 579
/RTAKEOVER for a Tracking Subsystem 579
Environments and Keywords 579
Usage. 579
/RTAKEOVER for an Active Subsystem 579
/RTAKEOVER for a Tracking Subsystem 580
Examples 580
Example for /RTAKEOVER Command at Active Site. 580
Example for /RTAKEOVER DUMPQ at Active Site 581
Example for /RTAKEOVER UNPLAN at Remote Site 581

Chapter 53. /SECURE 583
Format 583
Environments and Keywords 583
Usage. 583
Examples 584
Example 1 for /SECURE Command. 584
Example 2 for /SECURE Command. 585

Chapter 54. /SET 587
Format 587
Environments and Keywords 587
Usage. 587
Examples 588
Example 1 for /SET Command 588
Example 2 for /SET Command 588
Example 3 for /SET Command 588

Chapter 55. /SIGN 591
Format 591
Environments 591
Usage. 591
Examples 594
Example 1 for /SIGN Command 594
Example 2 for /SIGN Command 594
Example 3 for /SIGN Command 595

Chapter 56. /SMCOPY 597
Format 597
Environments and Keywords 597
Usage. 597
Example for /SMCOPY Command 598

Chapter 57. /SSR 599
Format 599

Environments 599
Usage. 599
Example for /SSR Command 599

Chapter 58. /START 601
Format 601
Environments and Keywords 604
Usage. 605
Examples 620
Example 1 for /START Command 620
Example 2 for /START Command 620
Example 3 for /START Command 620
Example 4 for /START Command 622
Example 5 for /START Command 622
Example 6 for /START Command 622
Example 7 for /START Command 622
Example 8 for /START Command 622
Example 9 for /START Command 623
Example 10 for /START Command 623
Example 11 for /START Command 623
Example 12 for /START Command 623
Example 13 for /START Command 623
Example 14 for /START Command 624
Example 15 for /START Command 624
Example 16 for /START Command 624
Example 17 for /START Command 624
Example 18 for /START Command 624
Example 19 for /START Command 625
Example 20 for /START Command 625
Example 21 for /START Command 625
Example 22 for /START Command 626
Example 23 for /START Command 626
Example 24 for /START Command 626
Example 25 for /START Command 626
Example 26 for /START Command 627

Chapter 59. /STOP 629
Format 629
Environments and Keywords 631
Usage. 632
Examples 644
Example 1 for /STOP Command 644
Example 2 for /STOP Command 644
Example 3 for /STOP Command 644
Example 4 for /STOP Command 644
Example 5 for /STOP Command 644
Example 6 for /STOP Command 645
Example 7 for /STOP Command 646
Example 8 for /STOP Command 646
Example 9 for /STOP Command 646
Example 10 for /STOP Command 646
Example 11 for /STOP Command 647
Example 12 for /STOP Command 647
Example 13 for /STOP Command 647
Example 14 for /STOP Command 647
Example 15 for /STOP Command 647

Example 16 for /STOP Command 648
Example 17 for /STOP Command 648
Example 18 for /STOP Command 649
Example 19 for /STOP Command 650
Example 20 for /STOP Command 651
Example 21 for /STOP Command 652
Example 22 for /STOP Command 652
Example 23 for /STOP Command 653
Example 24 for /STOP Command 653
Example 25 for /STOP Command 653
Example 26 for /STOP Command 654
Example 27 for /STOP Command 654
Example 28 for /STOP Command 655
Example 29 for /STOP Command 655
Example 30 for /STOP Command 655
Example 31 for /STOP Command 656
Example 32 for /STOP Command 656

Chapter 60. /SWITCH 657
Format 657
/SWITCH for an Active XRF Subsystem 657
/SWITCH for an Alternate XRF Subsystem 657
Environments and Keywords 657
Usage. 657
Examples 658
Example 1 for /SWITCH Command 658
Example 2 for /SWITCH Command 659

Chapter 61. TERMINATE 661
Format 661
Environments and Keywords 661
TERMINATE OLC 661
TERMINATE OLC Output Fields 662
Return, Reason, and Completion Codes for TERMINATE OLC 663
TERMINATE OLC Error Handling 666
Example for TERMINATE OLC Command 670

Chapter 62. /TEST 673
Format 673
Environments and Keywords 673
Usage. 673
Examples 675
Example 1 for /TEST Command 675
Example 2 for /TEST Command 675

Chapter 63. /TRACE 677
Format 677
Environments and Keywords 680
Usage. 681
Examples 693
Example 1 for /TRACE Command 693
Example 2 for /TRACE Command 694
Example 3 for /TRACE Command 694
Example 4 for /TRACE Command 694
Example 5 for /TRACE Command 695
Example 6 for /TRACE Command 695

Example 7 for /TRACE Command 695
Example 8 for /TRACE Command 695
Example 9 for /TRACE Command 695
Example 10 for /TRACE Command 696
Example 11 for /TRACE Command 696
Example 12 for /TRACE Command 696
Example 13 for /TRACE Command 696
Example 14 for /TRACE Command 697

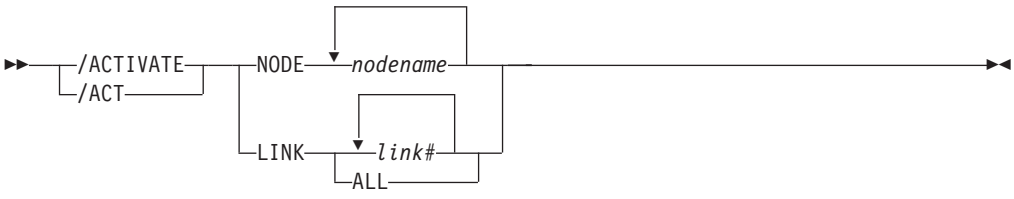
Chapter 64. /UNLOCK 699
Format 699
Environments and Keywords 699
Usage. 699
Examples 701
Example 1 for /UNLOCK Command. 701
Example 2 for /UNLOCK Command. 701
Example 3 for /UNLOCK Command. 701
Example 4 for /UNLOCK Command. 702
Example 5 for /UNLOCK Command. 702

Chapter 65. UPDATE 705
Format 705
Environments and Keywords 707
2 UPDATE AREA 709
2 UPDATE AREA Output Fields 711
2 Return, Reason, and Completion Codes for UPDATE AREA 711
2 The UPDATE AREA Command Compared to Other Commands 714
2 Examples for UPDATE AREA Command 714
| UPDATE DATAGRP 716
| UPDATE DATAGRP Output Fields 718
| Return, Reason, and Completion Codes for the UPDATE DATAGRP
| Command 719
| UPDATE DATAGRP Compared to Other Commands 720
| Examples for UPDATE DATAGRP Command 721
| UPDATE DB 723
| UPDATE DB Output Fields 729
| Return, Reason, and Completion Codes for the UPDATE DB Command 729
| UPDATE DB Compared to Other Commands 734
| Examples for UPDATE DB Command 735
UPDATE LE 738
UPDATE LE Output Fields 739
Return, Reason, and Completion Codes for UPDATE LE 740
Examples for UPDATE LE Command 740
UPDATE TRAN 742
2 UPDATE TRAN Output Fields 745
Return, Reason, and Completion Codes for UPDATE TRAN. 745
UPDATE TRAN Compared to Other Commands 747
Example 1 for UPDATE TRAN Command 748

Chapter 66. /VUNLOAD 751
Format 751
Environments and Keywords 751
Usage. 751
| Example for /VUNLOAD Command 752

Chapter 3. /ACTIVATE

Format



Environments and Keywords

Table 16 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 16. Valid Environments for the /ACTIVATE Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/ACTIVATE	X		X
LINK	X		X
NODE	X		X

Usage

/ACTIVATE is used to activate VTAM nodes or MSC links with a VTAM "VARY NET, ACTIVE" command. /ACTIVATE can also be used to undo the conditions set by the /IDLE command. /ACTIVATE is used with the IMS VTAM I/O Timeout Detection Facility.

LINK
Specifies the MSC links to be activated.

NODE
Specifies the VTAM nodes to be activated.

Examples

Example 1 for /ACTIVATE Command

Entry ET:

```
/ACTIVATE LINK 1
```

Response ET:

```
DFS058I  ACTIVATE COMMAND COMPLETED
```

Explanation: Link 1 has been activated.

Example 2 for /ACTIVATE Command

Entry ET:

```
/ACTIVATE NODE ABC
```

© Copyright IBM Corp. 1974, 2003

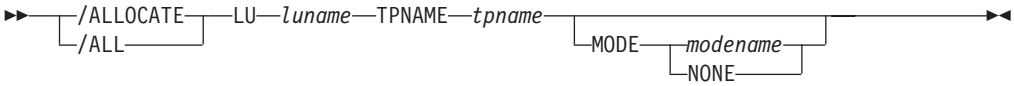
Response ET:

```
DFS058I  ACTIVATE COMMAND COMPLETED
```

Explanation: Node ABC has been activated.

Chapter 4. /ALLOCATE

Format



Environments and Keywords

Table 17 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 17. Valid Environments for the /ALLOCATE Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/ALLOCATE	X		X
LU	X		X
MODE	X		X
TPNAME	X		X

Usage

/ALLOCATE is a multisegment command similar to the /OPNDST command for VTAM terminals. It causes IMS to allocate a conversation to the specified LUNAME and TPNAME if any output is queued in IMS for that destination. If no such output exists, the command has no effect and (in a nonshared-queues environment) an error message is issued.

The /ALLOCATE command is used for delivering asynchronous output when the LU 6.2 device is available. The command can be originated by the remote LU 6.2 device for this purpose.

/ALLOCATE requires an end-of-message (EOM) indicator. An end-of-segment (EOS) indicator must be included for all segments preceding the last segment. For more information, see “Multisegment Command Input” on page 8

After the syntax checking of the /ALLOCATE command is successfully completed, the DFS058 ALLOCATE COMMAND COMPLETED message is issued and processing of the command continues asynchronously.

LU

Specifies the LU name of the LU 6.2 application program that is to be allocated. A network-qualified LU name is optional for the LUNAME keyword

MODE

identifies the LOGON MODE table entry that VTAM uses to determine the session operating characteristics. NONE, which can be used as a MODE keyword parameter, resets the MODE field to its null state.

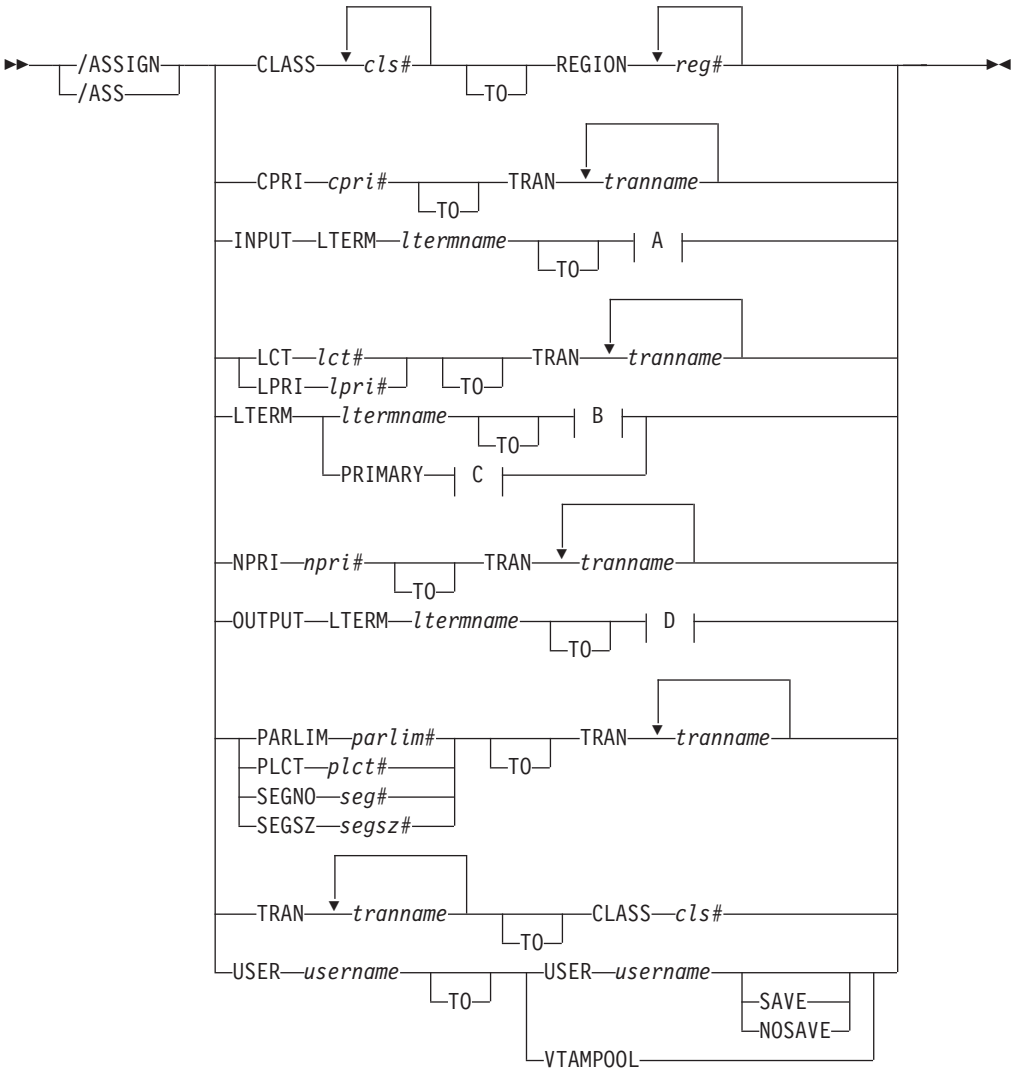
For more information on the MODE value, see *IMS Version 9: Administration Guide: Transaction Manager*.

© Copyright IBM Corp. 1974, 2003

TPNAME
Specifies the TP name of the LU 6.2 application program that is to be allocated.

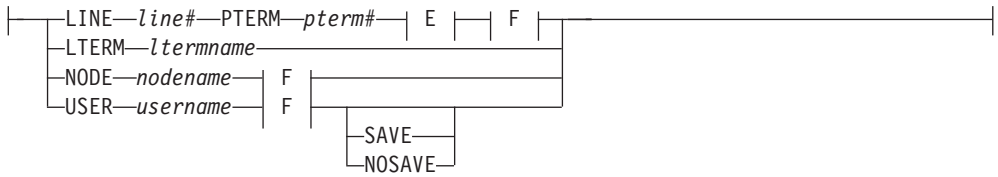
Chapter 5. /ASSIGN

Format



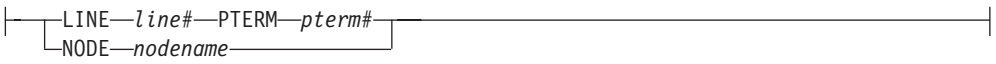
A:

B:



© Copyright IBM Corp. 1974, 2003

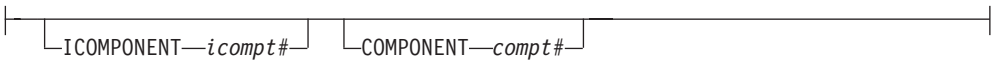
C:



D:

E:

F:



Environments and Keywords

Table 18 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 18. Valid Environments for the /ASSIGN Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/ASSIGN	X		X
CLASS	X		X
COMPONENT	X		X
CPRI	X		X
ICOMPONENT	X		X
INPUT	X		X
LINE	X		X
LCT	X		X
LPRI	X		X
LTERM	X		X
NODE	X		X
NOSAVE	X		X
NPRI	X		X
OUTPUT	X		X
PARLIM	X		X
PLCT	X		X
PTERM	X		X
REGION	X		X

Table 18. Valid Environments for the /ASSIGN Command and Keywords (continued)

Command / Keywords	DB/DC	DBCTL	DCCTL
SAVE	X		X
SEGNO	X		X
SEGSZ	X		X
TRAN	X		X
USER	X		X
VTAMPOOL	X		X

Usage

/ASSIGN alters the relationships between various IMS resources and resets certain values specified during IMS system definition.

Most changes made by the /ASSIGN command remain in effect until changed with another /ASSIGN command, or an IMS cold start (/NRESTART CHECKPOINT 0) is performed.

When dynamic LTERMs are assigned to dynamic users with the /ASSIGN LTERM TO USER command, or dynamic users are assigned to dynamic users with the /ASSIGN USER TO USER command, changes remain in effect only if the following conditions are true:

- The SAVE keyword is used.
- SAVE and NOSAVE keywords are omitted, but ASSNCHANGE=SAVE is specified in the DFSDCxxx PROCLIB member.

Otherwise, changes are lost when the destination user is deleted during user sign-off or IMS checkpoint.

Changes that affect regions, such as CLASS, are only in effect until the region terminates. For information on changing multiple systems assignments, see Chapter 36, “/MSASSIGN,” on page 427

Recommendation: Use the /ASSIGN command consistently on all IMS subsystems that are members of a VTAM generic resources group or share IMS queues. For example, assign an LTERM to a specific node for all IMS subsystems; assigning one LTERM to different nodes on different IMS subsystems could cause unpredictable results.

The /ASSIGN command is mirrored on an XRF alternate environment unless the assignment involves the primary or secondary master terminal, or the assignment of a CLASS to a REGION.

/ASSIGN commands attempting to assign LTERMs or users between static and dynamic terminals receive a unique error message. An error message is issued if /ASSIGN attempts any of the following

- To assign a dynamic LTERM to a static USER.
- To assign a static USER to a dynamic USER.
- To assign a dynamic USER to a static USER.
- To assign a dynamic LTERM to a LINE or NODE.
- To assign a static LTERM to a dynamic USER.

Restriction: You cannot use the /ASSIGN command to assign a user structure to a node that does not exist. Use the /OPN NODE nodename USER username command instead.

/ASSIGN in an IMSplex

In an IMSplex, the /ASSIGN LTERM and /ASSIGN USER commands alter the relationships between one or more logical terminals and a physical terminal or user.

Static Terminals

For static terminals in an IMSplex, the control blocks should exist in all the IMS systems where a user could possibly log on to that terminal. If /ASSIGN LTERM TO NODE is specified in an IMSplex, it should be issued from the Single Point Of Control (SPOC) application. If ROUTE is specified, it should be specified as ROUTE(*). Otherwise, the /ASSIGN command may not be processed consistently across the IMSplex and could cause unpredictable results. If one of the /ASSIGN commands fails, that IMS will send back a completion code to the Operations Manager (OM) indicating failure. It will be up to the IMS installation to correct that problem.

/ASSIGN LTERM TO NODE assigns the input and output capabilities of a logical terminal to a static VTAM node. The logical terminal may not be in conversational, response, or preset destination mode. The source and destination terminals do not need to be stopped or logged off.

If global resource information is not kept in the Resource Manager (RM) and the node exists locally, this command applies locally. If global resource information is kept in RM, this command applies globally, if the control blocks exist in RM. However, the assignment changes are not considered significant, and if the resource is deleted, the assignment changes are deleted as well. An LTERM that exists in the RM could be deleted if the static node to which it is assigned does not exist in the RM. If NODEA and LTERMA exist in the RM because the node is in exclusive mode, but static NODEB is not in RM, an /ASSIGN LTERM LTERMA NODE NODEB command would result in LTERMA being deleted in RM.

The IMS where the LTERM and node are active, or the command master if the resources are not active, will make the changes in the RM. In this case, ″active″ includes the case where the terminal is logged off, but has an RM affinity to a particular IMS. For /ASSIGN LTERM TO NODE, the LTERM and node might not be active in different IMSs.

When the log on of a static node attempts to write its LTERM names to the RM, but one or more of the LTERM names are already in use, IMS will reject the log on. This error should occur only if there are inconsistent definitions, or there have been inconsistent /ASSIGN commands issued in the IMSplex.

Dynamic Terminals

/ASSIGN LTERM TO USER assigns the input and output capabilities of an ETO logical terminal to an ETO user or a static ISC logical terminal to a static ISC subpool. The logical terminal may not be in conversational, response, or preset destination mode. The source user associated with the ETO LTERM might not be signed on. The destination ETO user does not have to be stopped. If global resource information is not kept in RM and the user exists locally, this command applies locally. If global resource information is kept in RM, this command applies globally (if the control blocks exist in RM). However, the assignment changes are not considered significant, and if the resource is deleted, the assignment changes are deleted as well. The IMS where the LTERM and node are active, or the command master if the resources are not active, will make the changes in the RM.

Use /ASSIGN LTERM TO USER with the SAVE keyword in an RM environment, to cause the assignment to persist and to apply the change to future IMSs that join the IMSplex. The SAVE keyword is valid only for dynamic terminals, so this persistence function is not available for static terminals. Use /ASSIGN LTERM TO USER with the NOSAVE keyword in an RM environment, to stop applying the change to future IMSs that join the IMSplex.

/ASSIGN USER TO USER assigns all the logical terminals associated with the source user to the destination user. The destination user, if dynamic, does not have to be stopped. If global resource information is not kept in RM and the user exists locally, this command applies locally. If global resource information is kept in RM, this command applies globally. Use /ASSIGN USER TO USER with the SAVE keyword in an RM environment to cause the assignment to persist and to apply the change to future IMSs that join the IMSplex. The SAVE keyword is valid only for dynamic terminals, so this persistence function is not available for static terminals. Use /ASSIGN USER TO USER with the NOSAVE keyword in an RM environment to stop applying the change to future IMSs that join the IMSplex.

The /ASSIGN command is allowed when the destination user does not exist. However, when resource information is being kept in RM, unless the SAVE keyword is used, this command will not be allowed. Assignments are not considered significant without the SAVE keyword, and if the user doesn't already exist due to some other significant status, there is no reason to create the user.

IMS will dynamically create the source LTERM or user (if it does not exist) to enable the /ASSIGN command. If the destination also does not exist, the SAVE keyword must be specified on the /ASSIGN command. Otherwise, the SAVE keyword is optional.

BTAM and VTAM

IMS supports the assignment of an LTERM between BTAM and VTAM terminals. However, IMS will not save any BTAM status. For example, if an LTERM is assigned from VTAM to BTAM, the LTERM will be deleted from the RM, and any status that might have been associated with that LTERM will no longer be recoverable.

Keywords and Parameters

CLASS

Specifies either the classes of transactions that a message processing region can schedule or the message processing class of a transaction.

The /ASSIGN CLASS cls# (TO) REGION reg# command can include up to four class parameters and replaces the class values that existed previously. Valid class parameters are numeric values from 1 to 999.

For static transactions, the CLASS parameter (cls#) is initially set by system definition using the TRANSACT macro statement. For CPI communications driven transactions, the initial value is obtained from the TP profile. For more information on the CPI Communications interface transactions, see *IMS Version 9: Administration Guide: Transaction Manager*.

COMPONENT

Allows output for a logical terminal to be directed to a specific component of a physical terminal. The COMPONENT parameter, which can have values 1, 2, 3 or 4, indicates the terminal component that should receive the output. Parameter values other than 1 are valid for the 3275 and SLU 1 terminals, as well as type 1 and type P secondary logical units. When assigning a

COMPONENT value from a terminal with components to a terminal without components, give the compt# a value of 1.

Table 19 shows the relationships between COMPONENT values and terminals. The first column lists the terminal, the second and third columns list the COMPONENT parameters that can have values of 1, 2, 3, or 4.

Table 19. Relationships Between COMPONENT/ICOMPONENT Values and Terminals

Terminal	1	2/3/4
3275	Video	Printer
3770	Console/Printer	Defined in TERMINAL macro on logon descriptor during system definition.
SLU 1	Defined in TERMINAL macro on logon descriptor during system definition.	
SLU P terminal	Defined in TERMINAL macro on logon descriptor during system definition.	
SLU 4	Defined in TERMINAL macro during system definition.	

CPRI
Specifies a new value for the current priority of a transaction. The CPRI keyword is not allowed for BMP transactions, since BMP transactions should always have a priority of 0. The new CPRI value takes effect the next time the transaction is scheduled. Valid CPRI parameters are numeric values from 0 to 14.

ICOMPONENT
Specifies the input component of a logical terminal to be directed to a specific component of a physical terminal. The ICOMPONENT parameter, which can have values 1, 2, 3, or 4, indicates the terminal component that should receive the input. Parameter values other than 1 are valid for the 3275 and SLU 1 terminals, as well as type 1 and type P secondary logical units. When assigning from a terminal with components to a terminal without components, give icompt# a value of 1. See Table 19 for relationships between ICOMPONENT values and terminal components.

INPUT
Specifies that the input capability of a logical terminal is to be assigned to a given nonswitched physical terminal without affecting the output assignment. More than one physical terminal can be assigned as input for a logical terminal, but the logical terminal used must be the first logical terminal in a “chain.”

LINE
Specifies the BTAM line to which the input or output capabilities (or both) of a logical terminal are to be assigned. The physical terminal must not be on a switched line, in conversational, response, or preset destination modes.

LCT
Specifies a new value for the limit count of a transaction. The new LCT value takes effect during execution of the transaction. Valid LCT parameters are numeric values from 1 to 65535.

LPRI
Specifies a new value for the limit priority of a transaction. The LPRI keyword is not allowed for BMP transactions, since BMP transactions should always have a priority of 0. The new LPRI value takes effect the next time the transaction is scheduled. Valid LPRI parameters are numeric values from 0 to 14.

LTERM

LTERM assigns the input and output capabilities of a logical terminal to the given nonswitched physical terminal or terminals.

If you specify `/ASSIGN LTERM` with LINE/PTERM selected, the first physical terminal (LINE/PTERM specification) is the input assignment. The second physical terminal (LINE/PTERM specification) is the output assignment. If there is no second physical terminal specification, then the first physical terminal specification is also the output assignment.

If you specify `/ASSIGN LTERM [TO] NODE`, the node specification is the input and output assignment.

No response is sent to a terminal referenced in an `/ASSIGN` command. The logical terminal must not be an inquiry logical terminal and must not have queueing or dequeuing of messages in progress.

LTERMs created dynamically can be assigned to dynamically created users. Static logical terminals can be assigned to other static terminals. Dynamic logical terminals cannot be assigned to lines, static nodes, dynamic nodes, or static users. Static logical terminals cannot be assigned to dynamic nodes or users. `/ASSIGN` commands that split the logical terminal so that the input LTERM is associated with one node and the output is associated with another node are not allowed for any ACF/VTAM terminals.

Physical Terminal Considerations

The physical terminal named in the command or indirectly referred to by being associated with a logical terminal in the command must not be on a switched line, in conversational, response, or preset destination modes. If a user logical terminal is specified, no logical terminal in the subpool can be currently signed on, and the line and physical terminal for input and output must be the same. If message processing programs are dependent upon a particular LTERM-to-component assignment, this relationship must be maintained by the operator.

MSGDEL processing for an LTERM is based on how the MSGDEL option was defined:

- For static LTERMs defined during system definition as part of the VTAMPOOL for ISC support and for dynamic LTERMs, the MSGDEL options of the users must match when moving LTERMs between the users.
- For all other static LTERMs, the MSGDEL option is defined in the TERMINAL macro for the associated physical terminal.
- If a terminal's MSGDEL capabilities are defined in the TERMINAL macro as MSGDEL=NONIOPCB, assignment of an LTERM can take place only if the LTERM's message queues are empty, except when the LTERM is assigned to a terminal also defined as MSGDEL=NONIOPCB.
- If a terminal's MSGDEL capabilities are defined as MSGDEL=SYSINFO or NOTERM, assignment of an LTERM can take place only if the LTERM's system message queue is empty, except when the LTERM is assigned to a terminal defined as MSGDEL=SYSINFO or NOTERM, or MSGDEL=NONIOPCB.

Master Terminal Considerations

The primary and secondary master terminals cannot be assigned to a user, an input-only or output-only device, a terminal in response mode, an unattended type 1 secondary logical unit terminal, an ISC node, or NTO terminal, or dynamic node. The LINE/PTERM or NODE to which the master

terminal is to be assigned must be started and in an operable state. The line and physical terminal for input and output must be the same.

When a 3270 is designated as the master terminal during IMS system definition, two master logical terminals are generated. One, referred to as the primary master terminal, must be assigned to a 3270 display (3275/3276/3278/3279) for all input and output messages. The other, referred to as the secondary master terminal, must be assigned to a 3270 printer (3284/3286/3287/3288/3289) for certain IMS-selected output messages.

To move a single master logical terminal from one physical terminal to another physical terminal (for example, the secondary master from one 3284 to another 3284, or the primary master from one 3277 to another 3277) without moving the other, use the formats of `/ASSIGN LTERM` command with one LINE/PTERM or NODE, and specify the LTERM name of the master terminal.

You can use either of the following commands:

- `/ASSIGN LTERM ltermname (TO) NODE nodename`
- `/ASSIGN LTERM ltermname (TO). LINE line# PTERM pterm#`

PRIMARY Parameter

The reserved parameter PRIMARY can be used with the LTERM keyword to change both the primary and secondary master terminals assignments to other physical terminals concurrently. The command formats used with the PRIMARY parameter are:

- `/ASSIGN LTERM PRIMARY (TO) LINE line#1 PTERM pterm#1/NODE nodename#1`

This command format assigns both the input and output of the primary master terminal to `LINE line#1 PTERM pterm#1`, and both the input and output of the secondary master terminal to NODE nodename.

- `/ASSIGN LTERM PRIMARY (TO) (LINE line#1 PTERM pterm#1/NODE nodename1)(LINE line#2 PTERM pterm#2/NODE nodename2)`

This command format assigns the input and output of the primary master terminal to the `LINE line#1 PTERM pterm#1/NODE nodename #1` and assigns the input and output of the secondary terminal to `LINE line#2 PTERM#2/NODE nodename#2`.

These two formats can reassign:

- The 3270 primary and 3270 secondary master terminals to different 3270 physical terminals, where pterm#1 or nodename1 is a 3270 display and pterm#2 or nodename2 is a 3270 printer.
- A 3270 master terminal complex to a non-3270 master terminal, where pterm#1 or nodename1 is a 2740.
- A non-3270 master terminal to a 3270 display and 3270 printer, where pterm#1 or nodename1 is a 3270 display and pterm#2 or nodename2 is a 3270 printer.

The first LINE/PTERM or NODE specifies the display device to which the primary master logical terminal is to be assigned. The second LINE/PTERM or NODE specifies the printer device to which both the input and output capabilities of the secondary are to be assigned. If only one LINE/PTERM or NODE is specified, then the input and output capabilities of the primary

master logical terminal and the secondary master logical terminal are assigned to the same LINE/PTERM or NODE.

Data Considerations

IMS does not edit data destined for a logical terminal when an /ASSIGN LTERM or /ASSIGN USER command is executed that affects physical terminal to logical terminal relationships. Ensure that data that is to be sent to a given physical terminal is suitable for transmission to a different physical terminal. For example, a print line (segment) 144 characters long destined for a 2780 terminal with a terminal 144-character printer will not print correctly if the logical terminal is assigned to a 2740 terminal with a maximum line size of 130 characters. If the Message Format Service (MFS) is used and the FMT definition included the second device, the change of physical terminal from one MFS-supported device to another MFS-supported device will produce correct output.

For a description of logical terminals (LTERMs), see *IMS Version 9: Administration Guide: System*. The command formats used with the /ASSIGN LTERM ltermname command are:

- /ASSIGN LTERM ltermname [TO] LINE line#1 PTERM pterm#1

This command format assigns the input and output capabilities of the logical terminal to LINE line#1 PTERM pterm#1. This command is not valid if the LTERM is dynamic.

```
/ASSIGN LTERM ltermname [TO] LINE line#1
PTERM pterm#1 LINE line#2
PTERM pterm#2
```

This command format assigns the input capability of the logical terminal to the first LINE/PTERM pair and assigns the output capability of the logical terminal to the second LINE/PTERM pair. This command is not valid if the LTERM is dynamic.

- /ASSIGN LTERM ltermname [TO] LINE line#1 PTERM pterm#1 PTERM pterm#2

This command format assigns the input capability of the logical terminal to LINE line#1 PTERM pterm#1 and assigns the output capability of the logical terminal to the same LINE, line#1, but to a different PTERM, pterm#2. This command is not valid if the LTERM is dynamic.

- /ASSIGN LTERM ltermname [TO] NODE nodenamez

This command format assigns the input and output capabilities of the logical terminal to NODE nodename. This command is not valid if the LTERM or NODE is dynamic.

- /ASSIGN LTERM ltermname1 [TO] LTERM ltermname2

This command format allows an input chain to be moved to a symbolic location, rather than to a specific line and physical terminal. The [TO] line and terminal are determined by the current assignment of the specified [TO] logical terminal. With this format, the output assignment is not affected. This form of the /ASSIGN command is invalid for ISC nodes defined for parallel sessions, and it is not supported for LTERMs associated with VTAM terminals.

- /ASSIGN LTERM ltermname [TO] USER username

This command format assigns the logical terminal to the given user. When moving logical terminals between users, the MSGDEL options of the users as defined during system definition or on user descriptor must match. The user related to LTERM ltermname must not be signed on in conversation, in response mode, or in preset mode. For 3600/FINANCE, SLU P, and ISC, the user (username) can remain allocated due to terminal message resynchronization requirements. However, the user cannot be signed on because the associated terminal session must be stopped and idle. The user can be allocated, but, if allocated, must be associated with a session that is stopped and idle. This command will turn off the DEADQ status for the user associated with LTERM ltermname.

NODE
Specifies the VTAM terminal to which the input and output capabilities of a logical terminal are to be assigned. The node must not be dynamic, or in conversational, response, or preset destination modes. If a session with a VTAM terminal is terminated after a message is sent but before the response has been received, message resynchronization is necessary for this terminal. The output message for which no response was received must remain associated with this terminal until message resynchronization determines when the terminal received the message. If you use the /ASSIGN command to move the message to a different terminal, message resynchronization is no longer possible.

NOSAVE
Indicates that the assignment changes specified by the `/ASSIGN` command are lost when control blocks are deleted by the system when they are no longer needed. Omitting SAVE and NOSAVE causes the system-wide default to be used, as specified by the ASSNCHANGE parameter in the DFSDCxxx PROCLIB member.

NPRI
Specifies a new value for the normal priority of a transaction. The NPRI keyword is not allowed for BMP transactions, because BMP transactions should always have a priority of 0. The new NPRI value takes effect the next time the transaction is scheduled. Valid NPRI parameters are numeric values from 0 to 14.

OUTPUT
Specifies that the output capability of a logical terminal is to be assigned to a given nonswitched physical terminal, without affecting the input assignment. OUTPUT is invalid for ISC nodes defined for parallel sessions.

PARLIM
Specifies a new value for the parallel processing limit count of a transaction. parlim# is the maximum number of messages that can currently be queued, but not yet processed, by each active message region currently scheduled for this transaction. An additional region will be scheduled whenever the transaction queue count (for shared queues environments, the successful consecutive GU count is used instead of the queue count) exceeds the PARLIM value multiplied by the number of regions currently scheduled for this transaction. Valid PARLIM parameters are numeric values from 0 to 32767 and 65535, where 65535 disables transaction load balancing.

/ASSIGN PARLIM is not valid for CPI Communications driven transactions.

PLCT
Specifies a new value for the processing limit count of a transaction. The

PLCT is the number of messages of this transaction code that a program can process in a single scheduling. The new PLCT values take effect the next time the transaction is scheduled. Valid PLCT parameters are numeric values from 0 to 65535.

/ASSIGN PLCT is not valid for CPI Communications driven transaction programs.

PTERM
Specifies the BTAM physical terminal to which the input or output capabilities (or both) of a logical terminal are to be assigned.

REGION
Specifies the message processing region being assigned new classes of transactions that the region can schedule.

SAVE
Prevents deletion of user and LTERM control blocks across session and IMS restarts. Control blocks will be retained until the NOSAVE keyword is used with the /ASSIGN command. Omitting SAVE and NOSAVE causes the system-wide default to be used, as specified by the ASSNCHANGE parameter in the DFSDCxxx PROCLIB member.

SEGNO
Creates or changes the limit on the number of application program output segments allowed in message queues for each GU call. Segment limits will be established by transaction code, thereby allowing specification of more than one value for each application program. The new SEGNO value takes effect during execution of the transaction. Valid SEGNO parameters are numeric values from 0 to 65535.

/ASSIGN SEGNO is not valid for CPI Communications driven transaction programs.

SEGSZ
Creates or changes the limit on the size of application program output segments allowed in message queues for each GU call. The new SEGSZ value takes effect during execution of the transaction. Valid SEGSZ parameters are numeric values from 0 to 65535.

/ASSIGN SEGSZ is not valid for CPI Communications driven transaction programs.

TRAN
Specifies the transaction being assigned to a message processing class or being assigned a value. The new class takes effect the next time the transaction is scheduled.

USER
Assigns logical terminals to ISC half-sessions or to dynamic users. When moving logical terminals between users, the MSGDEL options of the users must match. The MSGDEL option for the static ISC users is defined in the SUBPOOL macro during system definition. The MSGDEL option for dynamic terminals is defined in the OPTIONS keyword of the ETO USER descriptor. /ASSIGN USER to VTAMPOOL is not valid if USER is not ISC.

/ASSIGN LTERM ltermname (TO) USER username assigns the logical terminal to another USER. The user associated with LTERM ltermname must not be allocated. The USER username can be allocated, but it must be associated with a session that is stopped and idle, if allocated.

Note: Assigning an LTERM is not a permanent action. When the destination structure is deleted (for example, with /SIGN OFF) IMS discards information about the assignment. When the original dynamic user who owned that dynamic LTERM signs on again or gets a message switch, the dynamic LTERM is recreated for the original dynamic user.

If SAVE is specified, the dynamic user is not deleted unless another /ASSIGN command with the NOSAVE keyword is issued.

/ASSIGN USER username1 (TO) USER username2 assigns a string of logical terminals to another USER The first user (username1) in the command must contain at least one logical terminal and must not be signed on, in conversation, in response mode, or in preset mode. For 3600/FINANCE, SLU P, and ISC, the second user (username2) can remain allocated due to terminal message resynchronization requirements. However, the user cannot be signed on and associated terminal sessions must be stopped and idle. This command turns off the DEADQ status for USER username1.

In addition to the ISC rules and restrictions, following dynamic terminal restrictions also apply. The /ASSIGN USER to USER and /ASSIGN LTERM to USER commands are rejected if the source or destination dynamic user is in conversation mode, response mode, or preset mode. The second USER can be allocated, but it must be associated with a session that is stopped and idle, if allocated.

When existing dynamic LTERMs or dynamic users are assigned to a destination user that doesn't exist, the destination user is created unless it is rejected by the DFSINSX0 user exit.

VTAMPOOL

Is valid only for VTAM ISC sessions. It is used to force a cold start of an ISC session that cannot perform a successful restart.

/ASSIGN VTAMPOOL deallocates an entire string of logical terminals allocated to a given USER. The user must not be signed on, in conversation mode, response mode, or preset mode, and the terminal must be stopped and idle. This command is valid for static and dynamic ISC users.

Examples

Example 1 for /ASSIGN Command

Entry ET:

```
/ASSIGN CLASS 5 TO REGION 3
```

Response ET:

```
DFS058I  ASSIGN COMMAND COMPLETED
```

Explanation: Class 5 is assigned to region 3. Class 5 transactions are scheduled into region 3. This command resets any previous class assignments to this region.

Example 2 for /ASSIGN Command

Entry ET:

```
/ASSIGN CLASS 4 6 2A TO REGION 5
```

Response ET:

```
DFS058I  ASSIGN COMMAND COMPLETED EXCEPT CLASS 2A
```

Explanation: The requested assignment is complete except for 2A, which is an invalid class number.

Example 3 for /ASSIGN Command

Entry ET:

```
/ASSIGN CPRI 8 TO TRANSACTION PIT, SEED
```

Response ET:

```
DFS058I  ASSIGN COMMAND COMPLETED
```

Explanation: A current priority of 8 is set for the transactions named PIT and SEED.

Example 4 for /ASSIGN Command

Entry ET:

```
/ASSIGN INPUT LTERM JONES TO LINE 4 PTERM 3
```

Response ET:

```
DFS058I  ASSIGN COMMAND COMPLETED
```

Explanation: Logical terminal JONES is assigned to LINE 4 PTERM 3 for input identification and security. It associates the chain of logical terminals, of which LTERM JONES will be chained in first, with LINE 4 PTERM 3 for input. LTERM JONES must not be in the interior of an input chain (no other logical terminal can point to it). A physical terminal can point (for input only) to the first logical terminal of any chain. The output physical terminal for LTERM JONES is not changed.

Example 5 for /ASSIGN Command

Entry ET:

```
/ASSIGN LTERM APPLE TO LINE 5 PTERM 1
```

Response ET:

```
DFS058I  ASSIGN COMMAND COMPLETED
```

Explanation: Logical terminal APPLE has both its input and output capabilities assigned to LINE 5 PTERM 1. The components present on LINE 5 PTERM 1 must be compatible with the physical terminal previously related to logical terminal APPLE.

Example 6 for /ASSIGN Command

Entry ET:

```
/ASSIGN LTERM APPLE TO NODE JONES
```

Response ET:

```
DFS058I  ASSIGN COMMAND COMPLETED
```

Explanation: Logical terminal APPLE has both its input and output capabilities assigned to node JONES. The components present on node JONES must be compatible with the physical terminal previously related to logical terminal APPLE.

Example 7 for /ASSIGN Command

Entry ET:

```
/ASSIGN LTERM SMITH TO LINE 4 PTERM 6 PTERM 7 COMPONENT 2
```

Response ET:

```
DFS058I  ASSIGN COMMAND COMPLETED
```

Explanation: Logical terminal SMITH is assigned to LINE 4 PTERM 6 for input and LINE 4 PTERM 7 for output. Output for LINE 4 PTERM 7 is directed to COMPONENT 2.

Example 8 for /ASSIGN Command

Entry ET:

```
/ASSIGN LTERM X TO LINE 5 PTERM 7 PTERM 6 COMPONENT 4
ICOMPONENT 3
```

Response ET:

```
DFS058I  ASSIGN COMMAND COMPLETED
```

Explanation: Logical terminal X is assigned to LINE 5 PTERM 7 for input and to LINE 5 PTERM 6 for output. Input is only received from input component 3, while output is directed to component 4.

Example 9 for /ASSIGN Command

Entry ET:

```
/ASSIGN LTERM JONES TO LINE 4 PTERM 6 LINE 9 PTERM 1
```

Response ET:

```
DFS058I  ASSIGN COMMAND COMPLETED
```

Explanation: Logical terminal JONES is assigned to LINE 4 PTERM 6 for input capability and LINE 9 PTERM 1 for output capability. The component assignment is unaffected.

Example 10 for /ASSIGN Command

Entry ET:

```
/ASSIGN LTERM SMITH NODE JONES
```

Response ET:

```
DFS058I  ASSIGN COMMAND COMPLETED
```

Explanation: Logical terminal SMITH is assigned to node JONES for both input and output.

Example 11 for /ASSIGN Command

Entry ET:

```
/ASSIGN LTERM BROWN TO LTERM WHITE
```

Response ET:

```
DFS058I  ASSIGN COMMAND COMPLETED
```

Explanation: Logical terminal BROWN is removed from its present input chain (if one exists) of logical terminals and added to the end of the input chain (if one exists) of logical terminal WHITE. The output physical terminal for LTERM BROWN is not changed.

Example 12 for /ASSIGN Command

Entry ET:

```
/ASSIGN LTERM LAX USER ILL ICOMPONENT 1 COMPONENT 2
```

Response ET:

```
DFS058I  ASSIGN COMMAND COMPLETED
```

Explanation: Logical terminal LAX is reassigned from its existing user to the user ILL. The user associated with LAX cannot be allocated to an active session.

Example 13 for /ASSIGN Command

Entry ET:

```
/ASSIGN LTERM PRIMARY TO LINE 4 PTERM 3
```

Response ET:

```
DFS058I  ASSIGN COMMAND COMPLETED
```

Explanation: PTERM 3 on LINE 4 becomes the master terminal. PTERM 3 must be a 3270 display.

Example 14 for /ASSIGN Command

Entry ET:

```
/ASSIGN LTERM PRIMARY TO NODE BOSS
```

Response ET:

```
DFS058I  ASSIGN COMMAND COMPLETED
```

Explanation: The node, BOSS, becomes the primary master terminal and the secondary master terminal.

Example 15 for /ASSIGN Command

Entry ET:

```
/ASSIGN LTERM PRIMARY TO LINE 1 PTERM 2 LINE 2 PTERM 4
```

Response ET:

```
DFS058I  ASSIGN COMMAND COMPLETED
```

Explanation: PTERM 2 on LINE 1 becomes the primary master terminal and PTERM 4 on LINE 2 becomes the secondary master terminal. PTERM 2 is a 3270 display station and PTERM 4 is a 3270 printer.

Example 16 for /ASSIGN Command

Entry ET:

```
/ASSIGN SEGNO 50 TO TRANSACTION APPLE
```

Response ET:

```
DFS058I  ASSIGN COMMAND COMPLETED
```

Explanation: A limit of 50 output segments is set for the transaction APPLE.

Example 17 for /ASSIGN Command

Entry ET:

```
/ASSIGN SEGSZ 1000 TO TRANSACTION APPLE
```

Response ET:

```
DFS058I  ASSIGN COMMAND COMPLETED
```

Explanation: A maximum size of 1000 bytes is set for any one output segment of transaction APPLE.

Example 18 for /ASSIGN Command

Entry ET:

```
/ASSIGN USER ILL TO USER CAL
```

Response ET:

```
DFS058I  ASSIGN COMMAND COMPLETED
```

Explanation: All the logical terminals of user ILL are appended to the existing LTERM string of user CAL. Use of this form of the /ASSIGN command leaves ILL with no LTERMs and therefore unavailable for allocation to a session. Both user ILL and CAL cannot be allocated to a session.

Example 19 for /ASSIGN Command

Entry ET:

```
/ASSIGN USER CAL TO VTAMPOOL
```

Response ET:

```
DFS058I  ASSIGN COMMAND COMPLETED
```

Explanation: The entire allocated LTERM string defined for user CAL is reassigned to the VTAM pool. The node to which user CAL is allocated must not be in session and must be stopped and idle. User CAL is then available for allocation to any ISC session.

Example 20 for /ASSIGN Command

Entry ET:

```
/ASSIGN TRAN APPLE TO CLASS 5
```

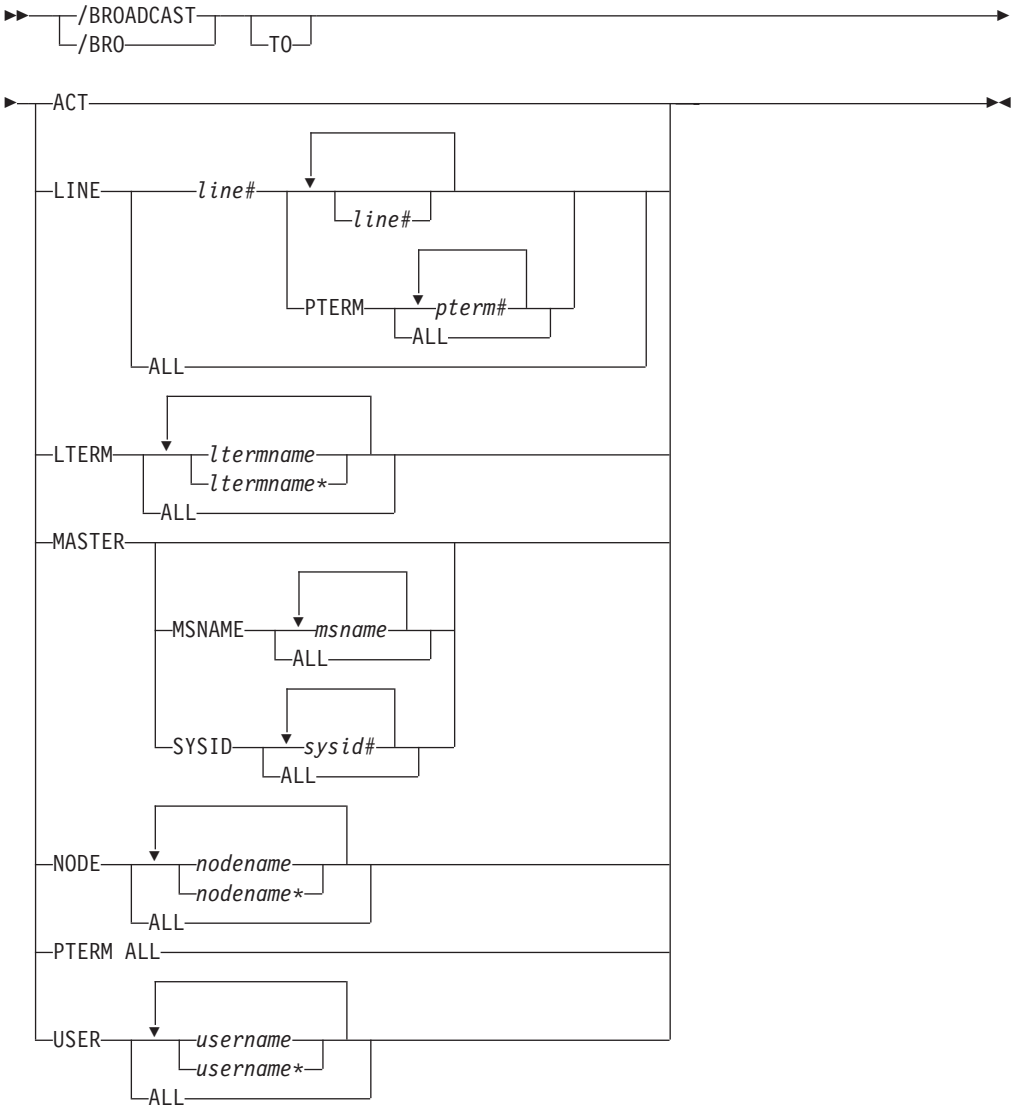
Response ET:

```
DFS058I  ASSIGN COMMAND COMPLETED
```

Explanation: The transaction named APPLE is assigned to class 5.

Chapter 6. /BROADCAST

Format



Environments and Keywords

Table 20 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 20. Valid Environments for the /BROADCAST Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/BROADCAST	X		X
ACT	X		X
LINE	X		X
LTERM	X		X

© Copyright IBM Corp. 1974, 2003

Table 20. Valid Environments for the /BROADCAST Command and Keywords (continued)

Command / Keywords	DB/DC	DBCTL	DCCTL
MASTER	X		X
MSNAME	X		X
NODE	X		X
PTERM	X		X
SYSID	X		X
USER	X		X

Usage

/BROADCAST is a multisegment command used to send a message to terminals in one or more IMS systems. For /BROADCAST commands entered by the master terminal operator, the multisegment input from this command is combined into 79-character segments for output transmission. The first input segment contains only the broadcast destination. The second and subsequent input segments must contain the data to be broadcast. Messages that are broadcast are sent even if the line, terminal, or both, are stopped at the time the broadcast is issued.

All /BROADCAST formats require an EOM indication to denote end-of-message; an EOS indication must be included for all segments that precede the last segment. See “Multisegment Command Input” on page 8 for more detail on using EOM and EOS.

Requirement: When the /BROADCAST command is issued from an MCS/E-MCS console, OM API, or an AOI application, a period must appear as a delimeter between the command and the message text.

Unlike message switches, broadcast messages are sent to a terminal even when the terminal, associated line, or both, are not available (stopped, process stopped, or locked). However, broadcast messages are never sent to terminals if the MSGDEL parameter of the TERMINAL macro is specified MSGDEL=NONIOPCB.

Broadcast messages are always queued for logical terminals. When the ACTIVE, LINE, NODE, PTERM, or USER keywords are used, IMS queues the message for the first logical terminal found that is associated for output purposes with the specified line or physical terminal.

ACT
Specifies that the supplied message is queued to the first LTERM allocated to each active node.

When operating on a dynamic terminal, the /BROADCAST ACT command only succeeds if a signed on user exists. A signed on user must exist, otherwise there is no destination to which to send the message.

LINE
Specifies that a message is to be sent to all terminals associated with the specified line. PTERM specifies that the message is to be sent to specific terminals on the associated line.

LTERM
Specifies that a message is queued to each named LTERM in the local system or any remote system. The remote logical terminal must be defined in the input system.

Where a LTERM does not exist, IMS attempts to create the LTERM and associated user structure if ETO is active. LTERM parameters can be generic, where the generic parameter specifies logical terminals that already exist.

When /BROADCAST LTERM ALL is specified, one copy of the message is queued for each logical terminal in the local system. In a multiple systems configuration, the message is not sent to remote systems when the ALL parameter is used. When more than one logical terminal is assigned to a physical terminal for output purposes, multiple copies of the message will result.

When the LTERM keyword specifies a logical terminal assigned to the VTAM pool, broadcast messages are queued for the first logical terminal in a subpool.

MASTER
Specifies that a message is to be sent to the IMS master terminal and to any specified secondary master terminal. Keywords SYSID and MSNAME can be used to further qualify the reserved parameter MASTER.

MSNAME
Specifies the logical link path in a multiple systems configuration.

NODE
Specifies that the supplied message is queued to the first output LTERM allocated to a terminal. If a terminal has no signed on user, no message can be queued. NODE parameters can be generic, where the generic parameter specifies nodes that already exist.

When operating on a dynamic terminal, the /BROADCAST NODE command only succeeds if a signed on user exists. A signed on user must exist, otherwise there is no destination to which to send the message.

PTERM
Specifies the physical terminal to which a message is to be sent.

SYSID
Specifies the system identification of a system in a multiple system configuration.

USER
Specifies the supplied message is queued to the first LTERM associated with an existing USER parameter can be generic. The /BROADCAST USER command applies only to existing dynamic users.

In an IMSplex, /BROADCAST USER queues the supplied message to the first LTERM associated with a dynamic user, if the user is signed on locally. /BROADCAST USER may not be used to queue a message to an LTERM associated with a user signed on to another IMS in the IMSplex, or not signed on at all.

Examples

Example 1 for /BROADCAST Command

Entry ET:

```
/BROADCAST ACTIVE (EOS)
SYSTEM WILL BE SHUTDOWN FOR PM (EOS)
IN 5 MINUTES (EOM)
```

Response ET:

```
DFS058I  BROADCAST COMMAND COMPLETED
```

Response RT:

```
SYSTEM WILL BE SHUTDOWN FOR PM IN 5 MINUTES
```

Explanation: The entered message is transmitted to all active terminals.

Example 2 for /BROADCAST Command

Entry ET:

```
/BROADCAST TO LTERM APPLE, TREE (EOS)
DON'T USE TRANSACTION GREENTRE UNTIL FURTHER (EOS)
NOTICE. (EOM)
```

Response ET:

```
DFS058I  BROADCAST COMMAND COMPLETED
```

Response RT:

```
DON'T USE TRANSACTION GREENTRE UNTIL FURTHER
NOTICE.
```

Explanation: The entered message is transmitted to the logical terminals named APPLE and TREE.

Example 3 for /BROADCAST Command

Entry ET:

```
/BROADCAST TO LINE ALL (EOS)
SYSTEM WILL SHUTDOWN AT 5PM (EOM)
```

Response ET:

```
DFS058I  BROADCAST COMMAND COMPLETED
```

Response RT:

```
SYSTEM WILL SHUTDOWN AT 5PM
```

Explanation: The entered message is transmitted to all physical terminals.

Example 4 for /BROADCAST Command

Entry ET:

```
/BROADCAST TO LINE 13 PTERM ALL (EOS)
EXPECT DEMO YOUR LINE AT 9PM (EOM)
```

Response ET:

```
DFS058I  BROADCAST COMMAND COMPLETED
```

Response RT:

```
EXPECT DEMO YOUR LINE AT 9PM
```

Explanation: The entered message is transmitted to all physical terminals on line 13.

Example 5 for /BROADCAST Command

Entry ET:

```
/BROADCAST MASTER SYSID 2 (EOS)
SYSTEM WILL SHUTDOWN AT 5:00 PM (EOM)
```

Response ET:

```
DFS058I  BROADCAST COMMAND COMPLETED
```

Response RT:

```
SYSTEM WILL SHUTDOWN AT 5:00 PM
```

Explanation: The message is transmitted to the master terminal of the system specified by the SYSID 2.

Example 6 for /BROADCAST Command

Entry ET:

```
/BROADCAST MASTER MSNAME BOSTON, CHICAGO (EOS)
SYSTEM WILL SHUTDOWN AT 5:00 PM (EOM)
```

Response ET:

```
DFS058I  BROADCAST COMMAND COMPLETED
```

Response RT:

```
SYSTEM WILL SHUTDOWN AT 5:00 PM
```

Explanation: The message is transmitted to the master terminals. of the remote systems specified by the MSNAMEs BOSTON and CHICAGO.

Example 7 for /BROADCAST Command

Remote Terminal entry:

```
/FORMAT DFSMO4
/BRO LTERM WTOR (eos)
this is segment 1 (eos)
this is segment 2 (eos)
this is segment 3 (eom)
```

Response ET:

```
DFS058I  BROADCAST COMMAND COMPLETED
```

Response RT:

```
THIS IS SEGMENT 1
THIS IS SEGMENT 2
THIS IS SEGMENT 3
```

Explanation: The remote terminal is first formatted by the /FORMAT command, where default format DFSMO4 supports the input of four segments. This is followed by /BROADCAST with four segments.

Chapter 7. /CANCEL

Format

Environments

Table 21 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command can be issued.

Table 21. Valid Environments for the /CANCEL Command

Command	DB/DC	DBCTL	DCCTL
/CANCEL	X		X

Usage

/CANCEL cancels all segments of a multisegment input message. It must be entered prior to the end-of-message (EOM) indicator from the terminal that was entering the message. Because a /CANCEL command must comprise a segment, it cannot be used to cancel a single-segment message.

On a non-3270 device, you can cancel a single-segment message by entering two asterisks (**), followed immediately by an end-of-segment (EOS) indicator. When MFS is used, you can define delete characters other than (**) to cancel other segments besides the first.

/CANCEL command has no meaning on display terminals where it is not possible to have some segments of a message already received by IMS while receiving subsequent segments.

Example for /CANCEL Command

Entry ET:

```
/BROADCAST TO ACTIVE (EOS)
SYSTEM WILL BE AVAILABLE (EOS)
```

Entry ET:

```
/CANCEL
```

Response ET:

```
DFS058I  CANCEL COMMAND COMPLETED
```

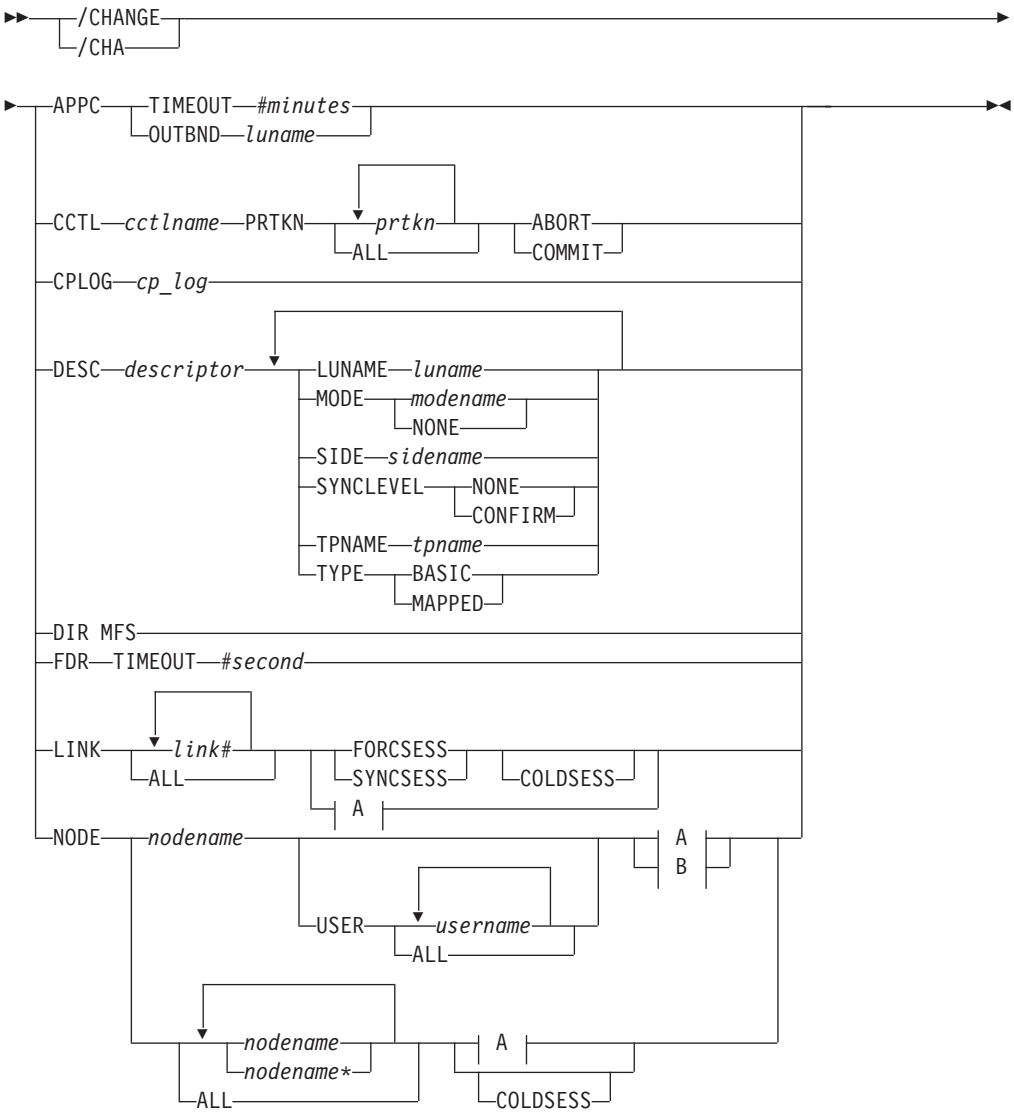
Explanation: All previously entered segments of the current message are discarded.

© Copyright IBM Corp. 1974, 2003

Chapter 8. /CHANGE

Format

/CHANGE Command: APPC Through NODE

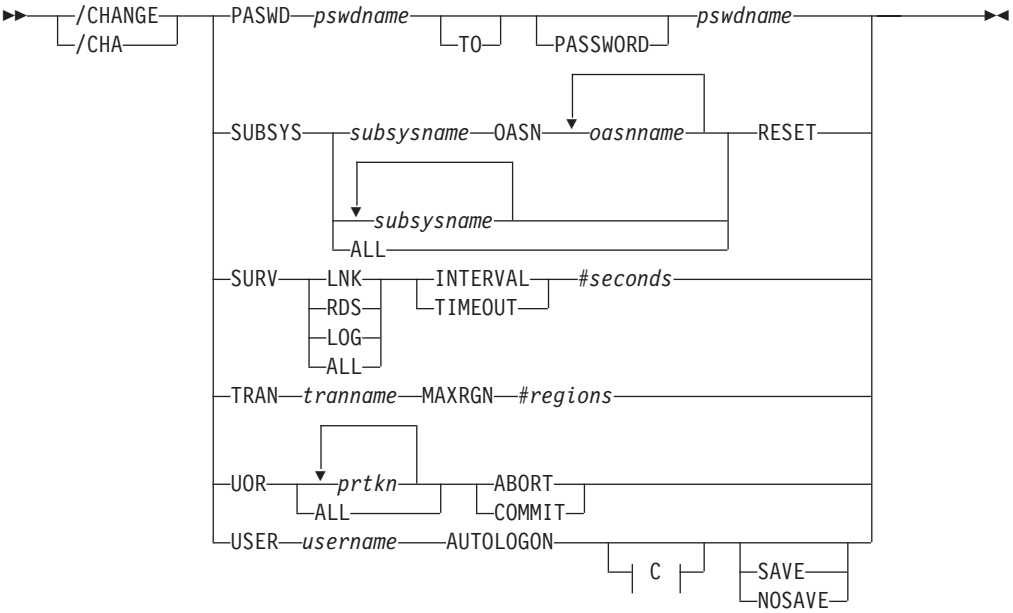


A:

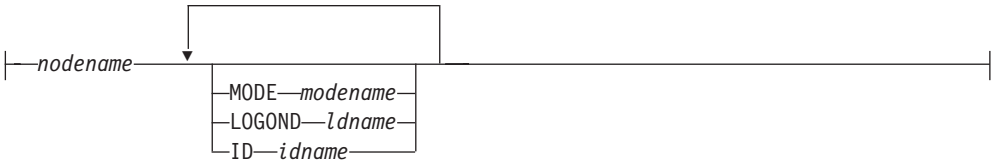
© Copyright IBM Corp. 1974, 2003

B:

/CHANGE Command: OUTBNDThrough USER



C:



Environments and Keywords

Table 22 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 22. Valid Environments for the /CHANGE Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/CHANGE	X	X	X
ABORT	X	X	
APPC	X		X
ASR	X		X
AUTOLOGON	X		X
CCTL	X	X	
COLDSESS	X		X

Table 22. Valid Environments for the /CHANGE Command and Keywords (continued)

Command / Keywords	DB/DC	DBCTL	DCCTL
COMMIT	X	X	
CPLOG	X	X	X
DESC	X		X
DIR	X		X
FDR	X	X	
FORCSESS	X		X
ID	X		X
INTERVAL	X		X
LINK	X		X
LOGOND	X		X
LUNAME	X		X
MAXRGN	X		X
MODE	X		X
NODE	X		X
NOSAVE	X		X
OASN	X	X	X
OUTBND	X	X	X
PASWD	X	X	X
RESET	X	X	X
SAVE	X		X
SIDE	X		X
SUBSYS	X	X	X
SURV	X		X
SYNCLEVEL	X		X
SYNCSESS	X		X
TIMEOUT	X	X	X
TPNAME	X		X
TRAN	X		X
TYPE	X		X
UOR	X	X	
USER	X		X

Usage

/CHANGE is a multisegment command used to change or delete internal resources within IMS.

All /CHANGE formats require an EOM indication to denote end-of-message; an EOS indication must be included for all segments that precede the last segment. See “Multisegment Command Input” on page 8 for more detail on using EOS and EOM.

APPC TIMEOUT
Specifies a change to the timeout value for APPC/IMS. This value is set in the DFSDCxxx member of IMS.PROCLIB. IMS passes this value to APPC/z/OS for every implicit APPC/IMS conversation.

The timeout value (*#minutes*) must be between zero and 1440. If the timeout value is zero, APPC/IMS timeout will be deactivated.

A `DFS34091` or `DFS3491I` message is issued after the `/CHANGE APPC TIMEOUT` command is issued to inform the operator of the new timeout value.

OUTBND
Specifies a different outbound LU. The specified LU must be one of the APPC LUs defined in the APPCPMxx member of the SYS1.PROCLIB library. The default outbound LU is BASE LU.

CCTL
Specifies the coordinator control subsystem. The recovery elements are resolved by IMS.

cctlname
Specifies CCTL subsystem ID.

prtkn
Specifies the pseudo recovery token, which designates the unit of recovery to be aborted or committed. Use the `/DISPLAY CCTL` command to determine the pseudo recovery token name.

ABORT
Backs out changes for a unit of recovery. After completion of backout, the recoverable indoubt structure (RIS) is removed.

COMMIT
Commits changes for a unit of recovery. After the process is complete, the RIS is removed.

CPLOG
Is used to change the value of the IMS execution parameter, CPLOG. The IMS CPLOG execution parameter specifies the number of system log records between system-generated checkpoints.

cp_log
This value must be specified as one or more numeric characters followed by either K or M. Values can range from 1K to 16M.

DESC
Specifies the name of an LU 6.2 descriptor that will be updated with new values. An error message is issued if the specified descriptor is not found.

Note: The `/CHANGE DESC` command changes the destination for future messages only. It does not the change the destination for existing messages. The existing output messages are delivered only to the luname and tpname that had been previously specified for that message. This restriction is required by security requirements so message delivery to the intended destination only occurs at the time the messages are created.

LUNAME
Specifies the LU name that is updated in the LU 6.2 descriptor.

The luname value in the descriptor is set to blanks unless the LUNAME keyword is also specified. A network-qualified LU name is optional for the LUNAME keyword.

MODE
Specifies the VTAM mode table entry name that is updated in the LU 6.2 descriptor. The NONE parameter resets the mode field to its null state.

The MODE value in the descriptor is set to blanks unless the MODE keyword is also specified.

SIDE
Specifies the APPC/z/OS side information that is set in the LU 6.2 descriptor. The side information contains default values for APPC conversation attributes such as LUNAME, TPNAME, and MODE.

SYNCLEVEL
Specifies the APPC sync level that is updated in the LU 6.2 descriptor. One of the following must be specified:

CONFIRM
IMS sync point processing continues.

NONE
IMS sync processing continues despite a session failure.

TPNAME
Specifies the tpname that is updated in the LU 6.2 descriptor. Message DFS182 is issued if the TPNAME parameter specified is DFSSIDE.

The TPNAME value in the descriptor is set to blanks unless the TPNAME keyword is also specified.

TYPE
Specifies the APPC conversation type that is updated in the LU 6.2 descriptor.

The conversation types are:

BASIC
Specifies that the data is to be formatted by the transaction programs, using the pattern “LL,data,LL,data”.

MAPPED
Specifies that the data is to be formatted by APPC.

DIR
Specifies that the entries in the MFS dynamic directory are to be deleted. This restores the dynamic directory to the original state that it was in just after IMS was initialized.

When IMS is initialized, IMS creates an MFS block primary directory based on the contents of all the $$IMSDIR members found. At the same time, IMS issues GETMAINs to acquire additional storage that is large enough to hold approximately 10% of the members in the active format library. While IMS is running, entries are only added to the dynamic directory, they are not deleted. Therefore, the dynamic directory continues to get larger until it runs out of space. When this happens, IMS can extend the amount of space in the dynamic directory to hold another 10%. IMS can extend the size of the dynamic directory a maximum of 9 times.

If you want to delete the in-storage (or index) entries from the dynamic directory without restarting IMS or performing an online change for the format library, use the /CHANGE DIR MFS command.

FDR
Specifies a change to the timeout value for IMS Fast Database Recovery surveillance. This value is set in the DFSFDRxx member of IMS.PROCLIB. IMS uses this value to determine how long to wait before initiating a Fast Database Recovery takeover.

The timeout value (*#seconds*) must be at least 3 but no greater than 999.

IMS rejects this command if the active IMS subsystem is not connected to a Fast Database Recovery region.

LINK
Applies only to IMS systems linked by MSC using VTAM.

Use the LINK keyword with FORCSESS, SYNCSESS, COLDSESS to override the system definition option defined for forcing resynchronization until the next /CHANGE LINK command or IMS cold start.

FORCSESS
Forces a session to come up, whether or not the message sequence numbers agree. When FORCSESS is used with COLDSESS, the session is set up so that cold start can occur, and, when it is cold started, it is forced to come up, whether or not the message sequence numbers agree. The use of FORCSESS could cause messages to be lost, so it should be used if a session could not be brought up, and loss of messages is of little importance while a session is being brought up.

SYNCSESS
Is used to complete session initiation, only if the message sequence numbers and the restart modes agree. When this keyword is used with COLDSESS, the session is set up so that it can be cold started, and, after cold start, it can be initiated only if the sequence numbers and the restart modes agree.

COLDSESS
When used in conjunction with the LINK keyword, COLDSESS enables a session to be cold started. If used with FORCSESS or SYNCSESS, the COLDSESS keyword is always processed first. Use of this keyword could cause messages to be lost. This keyword should be used if one of the systems associated with an MSC link goes down, and the only way to bring the session up is to cold start it.

The COLDSESS keyword can be specified to terminate control blocks associated with the specified link or links and reset the link to COLD. COLDSESS should be used only after the /PSTOP command is issued and completes against the link or links and the links appear to be hung. If COLDSESS is specified, it should be used on both sides of the link or links. Gather documentation to determine why the link or links did not come down normally.

ASR
Changes the automatic keyword session restart designation of a link. The default parameter for ASR is ON.

Automatic session restart is not necessarily activated for a link just because a status of ASR is displayed for that link. You must also have coded SONSCIP=YES on the APPL definition statement for VTAM when defining the network for the VTAM.

MODE

Changes the default mode table name of a link. This default is usually established by system definition. Parameter NONE resets this field to its null state (as if no mode table name was specified at system definition).

Changing ASR or mode table name for non-VTAM MSC links is invalid. If non-VTAM MSC links are referred to specifically in the `/CHANGE LINK ASR` or `/CHANGE LINK MODE` commands, they will be marked in error.

NODE

Specifies a VTAM node to be changed. The NODE parameter can be generic if the USER keyword is not present. The generic parameter specifies nodes that already exist.

ASR

The ASR keyword allows you to change the automatic session restart designation of a node. The default parameter for ASR is ON.

Automatic session restart is not necessarily activated for a node just because a status of ASR is displayed for that node. You must also have coded SONCSIP=YES on the APPL definition statement for VTAM when defining your network.

COLDSESS

When COLDSESS is used with the NODE keyword, it sets up the SLU P or FINANCE session so that the session can be cold started. The COLDSESS keyword should be used if the SLU P or FINANCE session has experienced problems and attempts to warm start the session fail. The COLDSESS keyword terminates terminal and user control blocks associated with the specified node or nodes, and resets the node status to COLD.

In an IMSplex, if global resource information is not kept in Resource Manager (RM), the change is applied locally. If global resource information is kept in RM, the change is applied globally.

A `DFS0581 COMMAND COMPLETE EXCEPT` message may be received if the node is temporarily in use by another task even if the criteria (node is terminated and idle) for successful completion of the command is met.

FORCSESS, SYNCSESS

The FORCSESS and SYNCSESS keywords are only valid for ISC nodes. Specify FORCESS and SYNCSESS to override the system definition or logon descriptor option defined to force or not force synchronization of sessions. This override is effective until the next `/CHANGE` command is issued or an IMS cold start is effected.

MODE

Changes the default mode table name of a node. This default is usually established by system definition or logon descriptor. MODE resets this field to its null state (as if no mode table name had been specified at system definition).

Changing ASR or the mode table name for VTAM 3270 nodes is invalid. If nodes of this type are referred to specifically in the `/CHANGE ASR` or `/CHANGE MODE` commands, they are marked in error.

USER

Indicates the ISC user that is allocated to the node or the dynamic user that is signed on to the node.

`/CHANGE USER AUTOLOGON SAVE` changes the autologon information previously specified by a user descriptor, output creation exit routine (DFSINSX0), or signon exit routine (DFSSGNX0). The autologon change is saved. The autologon information includes the node name, the mode table, the logon descriptor, and the ID. If the Resource Manager is active, the user will be dynamically created if it does not already exist. The SAVE keyword must be specified. If SAVE is not specified, the command is rejected with message DFS1199.

OUTBND

This keyword lets you change the current outbound LU. The default outbound LU is BASE LU. A different outbound LU can be specified with the OUTBND= parameter in the DFSDCxxx PROCLIB member. The specified LU must be one of the APPC LUs defined in the APPCPMxx member of the SYS1.PROCLIB library.

PASWD

Password security is created by the IMS Security Maintenance utility and invoked during IMS restart through the presence of the PASWD operand. PASWD *pswdname* must be defined prior to the entry of this command. Passwords are defined by use of the Security Maintenance utility or by a previous /CHANGE command. Command execution results in a direct password replacement. If the password operand pswdname does not exist in the user's IMS system, the /CHANGE command is rejected. Because duplicate passwords are not acceptable, PASWD pswdname (where this password name is the second name given in the command) must not have been previously defined when `/CHANGE PASWD` is entered or the password will be rejected.

SUBSYS

Specifies the subsystem name from which IMS recovery elements are to be deleted. OASN specifies the outstanding recovery element to be deleted. RESET causes an incomplete unit of work assigned to an external subsystem (not a CCTL subsystem) to be deleted.

SURV

Specifies that the interval or timeout value on the active or alternate system in an XRF environment is changed. The following keyword parameters identify the surveillance mechanism:

LNK	IMS ISC link
LOG	IMS system log
RDS	IMS restart data set
ALL	Same as specifying LNK, LOG, and RDS

The INTERVAL or TIMEOUT keyword identifies the new value in seconds and must be in the range 1 through 99. When `/CHANGE SURV` is entered on the active system, it becomes effective on both the active and alternate systems, if the alternate system is up. If entered on the alternate system, only the alternate system is changed.

It is recommended that surveillance be changed on the active system and allowed to take affect on the alternate system.

Additional considerations when changing the TIMEOUT values are shown in Table 23 Listed in the table are the TIMEOUT value requirements for certain systems where the command is entered on and the action taken for exceptions.

Table 23. Changing the TIMEOUT Value

System the Command is Entered on	Requirement	Action Taken for Exceptions
Active	New timeout value must be ≥ twice the active interval value.	DFS3832 issued. Active timeout value forced to twice the active interval value.
Alternate	Alternate interval value must be ≥ active interval value.	DFS3812 issued. Alternate interval value forced to active interval value.
Alternate	New timeout value must be ≥ twice the alternate interval value.	DFS3832 issued. Alternate timeout value forced to twice the alternate interval value.

Additional considerations when changing the INTERVAL values are shown in Table 24 Listed in the table are the INTERVAL value requirements for certain systems where the command is entered and the action taken for exceptions.

Table 24. Changing the INTERVAL Value

System the Command is Entered on	Requirement	Action Taken for Exceptions
Active	Twice the new interval value must be ≤ active timeout value.	DFS3832 issued. Active timeout value forced to twice the new interval value.
Alternate	Alternate interval value must be ≥ active interval value.	DFS3812 issued. Alternate interval value forced to active interval value.
Alternate	New alternate interval value must be ≥ alternate LOG interval value (LNK and RDS).	DFS3833 issued. Alternate interval value forced to alternate log interval value.
Alternate	New alternate interval value ≤ alternate RDS and LNK interval value (LOG only).	DFS3833 issued. Alternate interval value forced to alternate RDS then LNK interval value.
Alternate	Twice the new interval value must be ≤ alternate timeout value.	DFS3832 issued. Alternate timeout value forced to twice the new interval value.

TRAN, MAXRGN

The MAXRGN keyword is used with the TRAN keyword to change the maximum number of regions that can be simultaneously scheduled for a given transaction. The transaction must be eligible for parallel scheduling (load balancing). For static transactions, the MAXRGN parameter (*#regions*) is initially set by system definition using the TRANSACT macro statement. For CPI Communications driven transactions, the initial value is obtained from the TP profile. The value of *#regions* must be between 0 and the number specified on the *MAXPST= region* parameter.

Related Reading: For more information on the TP profile, see the *IMS Version 9: Administration Guide: Transaction Manager*. For more information about the *MAXPST=* parameter, see the *IMS Version 9: Installation Volume 2: System Definition and Tailoring*.

The /DISPLAY TRAN command indicates whether a transaction is eligible for load balancing with a status of BAL, followed (in parentheses) by the current maximum number of regions that can be simultaneously scheduled.

UOR
Specifies that IMS should resolve units of recovery (UORs) for protected resources on the RRS/z/OS recovery platform.

Recommendation: Use /CHANGE UOR only when you are certain that no other resource managers would be adversely affected after IMS resolves the UOR.

prtkn
Specifies the six-byte pseudotoken that designates the UOW to be committed. Use the /DISPLAY UOR command to obtain the *prtkn*.

If you specify ALL, the /CHANGE UOR command affects all units of recovery.

ABORT
Specifies that IMS back out changes for the protected resources.

COMMIT
Specifies that IMS make changes permanent for the protected resources.

USER
Specifies that an ETO will change. The user parameter cannot be generic.

AUTOLOGON
Specifies that the autologon information previously specified by a user descriptor, the output creation exit routine (DFSINSX0), or the signon exit routine (DFSSGNX0) is being updated dynamically.

nodename
Specifies the autologon terminal session for the specified user. Omitting the nodename clears all autologon information.

MODE
Specifies the VTAM mode table entry name.

LOGOND
Specifies the logon descriptor used to build the terminal control blocks.

ID Specifies the ISC partner's half-session qualifier (if the terminal is ISC).

NOSAVE
Indicates that the changed autologon information should not be retained. Deletion of the user occurs when an IMS checkpoint is taken, at session termination, IMS restart, or XRF takeover. If SAVE and NOSAVE are omitted, IMS uses the system-wide default as specified in the DFSDCxxx PROCLIB member.

SAVE
Indicates that the changed autologon information should be retained. This keyword prevents deletion of the user and remains in effect across a restart or XRF takeover until another /CHANGE command with the NOSAVE option is issued. If SAVE and NOSAVE are omitted, IMS uses the system-wide default as specified in the DFSDCxxx PROCLIB member.

If global resource information is kept in Resource Manager, the change is applied globally when the SAVE keyword is specified. If SAVE is not specified in this environment, the command is rejected.

Examples

Example 1 for /CHANGE Command

This set of examples shows that an INDOUBT unit of recovery can be aborted if the INDOUBT status cannot be resolved. The /CHANGE ... PRTKN command backs out changes made to the database.

Entry ET:

```
/DISPLAY CCTL CICS1 IND
```

Response ET:

```
CCTL     PSEUDO-RTKN RECOVERY-TOKEN   REGID  PSBNAME   STATUS
CICS1                                                  ATTACHED
         000100C0    9FFA956B7AE24E00        BMP255    INDOUBT
         00010040    9FFA9568FF594301        BMP255    INDOUBT
 *90067/113446*
```

Entry ET:

```
/CHANGE CCTL CICS1 PRTKN 000100C0 ABORT (EOM)
```

Response ET:

```
DFS058I  CHANGE COMMAND COMPLETED
DBS0699I REYNC ABORT COMPLETE FOR PSB BMP255...
```

Explanation: The INDOUBT unit of recovery whose pseudo recovery token (PRTKN) is 000100C0 has been aborted.

Example 2 for /CHANGE Command

This set of commands illustrate how the ASR setting of a link can be modified by the /CHANGE command:

Entry ET:

```
/DISPLAY LINK 6
```

Response ET:

```
LINK PARTNER   RECD  ENQCT  DEQCT   QCT   SENT
   6 AL          0      0      0     0      0 ASR PSTOPPED IDLE COLD
*90179/102004*   IMSA
```

Explanation: Automatic Session Restart was defined for link 6 by system definition.

Entry ET:

```
/CHANGE LINK 6 ASR OFF (EOM)
```

Response ET:

```
DFS058I  CHANGE COMMAND COMPLETED
```

Entry ET:

```
/DISPLAY LINK 6
```

Response ET:

```
LINK PARTNER   RECD  ENQCT  DEQCT    QCT    SENT
   6 AL          0      0      0      0       0 PSTOPPED IDLE COLD
*90179/102126*
```

Explanation: Automatic Session Restart is not available for link 6

Entry ET:

```
/CHANGE LINK 6 ASR (EOM)
```

Response ET:

```
DFS058I  CHANGE COMMAND COMPLETED
```

Explanation: The ASR parameter will default to ON.

Entry ET:

```
/DISPLAY LINK 6
```

Response ET:

```
LINK PARTNER   RECD  ENQCT  DEQCT    QCT    SENT
   6 AL          0      0      0      0       0 ASR PSTOPPED IDLE COLD
*90179/102300*
```

Explanation: Automatic Session Restart is in effect again as a result of the second /CHANGE command.

Example 3 for /CHANGE Command

Changing ASR for NODE or NODE/USER is similar to changing ASR for LINK. See the previous example for the commands used to change ASR.

Example 4 for /CHANGE Command

Entry ET:

```
/DISPLAY NODE LUTYPEP1 MODE
```

Response ET:

```
NODE-USR   TYPE   DEF MODETBL  ACT MODETBL
 LUTYPEP1  SLUP   DEFRESP
 *90179/100630*
```

Explanation: DEFRESP is the mode table name defined for node LUTYPEP1 at system definition or logon descriptor or resource creation. The session is not active so the active mode table field (ACT MODETBL) is blank.

Entry ET:

```
/CHANGE NODE LUTYPEP1 MODE XXXXXXXX (EOM)
```

Response ET:

```
DFS058I CHANGE COMMAND COMPLETED
```

Entry ET:

```
/DISPLAY NODE LUTYPEP1 MODE
```

Response ET:

```
NODE-USR TYPE   DEF MODETBL  ACT  MODETBL
LUTYPEP1 SLUP   XXXXXXXX
*84179/100733*
```

Explanation: The default mode table name has been changed to XXXXXXXX by the previous /CHANGE command. The session is still not active so the active mode table field (ACT MODETBL) is blank.

Example 5 for /CHANGE Command

Entry ET:

```
/DISPLAY NODE LUTYPEP
```

Response ET:

```
NODE-USR  TYPE   CID      RECD  ENQCT  DEQCT  QCT  SENT
LUTYPEP   SLUP  00000000    37     37     37    0    37  IDLE
*98276/153630*
```

Explanation: Node LUTYPEP is terminated warm after session received and processed 37 messages.

Entry ET:

```
/CHANGE NODE LUTYPEP COLDSESS (EOM)
```

Response ET:

```
DFS058I CHANGE COMMAND COMPLETED
```

Entry ET:

```
/DISPLAY NODE LUTYPEP
```

Response ET:

```
NODE-USR  TYPE   CID      RECD  ENQCT  DEQCT  QCT  SENT
LUTYPEP   SLUP  00000000     0      0      0    0     0  IDLE COLD
*98279/153630*
```

Explanation: Terminal and user blocks associated with node LUTYPEP have been cleared and the status has been reset to COLD.

Example 6 for /CHANGE Command

Entry ET:

```
/CHANGE PASSWORD 1234 TO WXYZ (EOM)
```

Response ET:

```
DFS058I  CHANGE COMMAND COMPLETED
```

Explanation: Password 1234 is changed to password WXYZ.

Example 7 for /CHANGE Command

Entry ET:

```
/CHANGE SUBSYS DSN RESET (EOM)
```

Response ET:

```
DFS058I  CHANGE COMMAND COMPLETED
```

Explanation: Reset all IN-DOUBT recovery units for subsystem DSN.

Example 8 for /CHANGE Command

Entry ET:

```
/CHANGE SUBSYS ALL RESET (EOM)
```

Response ET:

```
DFS058I  CHANGE COMMAND COMPLETED
```

Explanation: Reset all IN-DOUBT recovery units for all subsystems.

Example 9 for /CHANGE Command

Entry ET:

```
/CHANGE SUBSYS ABCD OASN 99 685 2920 RESET (EOM)
```

Response ET:

```
DFS058I  CHANGE COMMAND COMPLETED
```

Explanation: Reset IN-DOUBT recovery units identified by OASN numbers 99, 685, 2920 for subsystem ABCD.

Example 10 for /CHANGE Command

This set of commands illustrates how SURVEILLANCE can be modified by the /CHANGE command.

Entry ET (Alternate system):

```
/DISPLAY HSB
```

Response ET (Alternate system):

```
RSENAME       STATUS    PHASE    IMS-ID    VTAM UVAR  ACT-ID    LOG-TIME
DFSRSENM      BACKUP    TRK      IMSB      USERVAR    IMSA      10:35:17
              BACKUP SYSTEM                ACTIVE SYSTEM
SURVEILLANCE  INTERVAL  TIMEOUT  STATUS     INTERVAL  TIMEOUT  STATUS
LOG                2       99  INACTIVE          2       99  INACTIVE
LNK                4       99  INACTIVE          4       99  INACTIVE
RDS                3       99  INACTIVE          3       99  INACTIVE
TAKEOVER CONDITIONS - ALARM AUTO
RDS LINK LOG                  *RDS LINK
VTAM                          *IRLM
*91226/103517*
```

Entry ET (Alternate system):

```
/CHANGE SURVEILLANCE LNK INTERVAL 3 (EOM)
```

Response ET (Alternate system):

```
DFS058I CHANGE COMMAND COMPLETED
DFS3812I BACKUP LNK INTERVAL VALUE OF 3 HAS BEEN FORCED TO ACTIVE VALUE OF 4
DFS3811I LNK SURVEILLANCE INACTIVE: INTERVAL VALUE CHANGED FROM  4 TO  4
```

Explanation: An attempt to change the interval value was made by entering /CHANGE on the alternate system. The new alternate interval value was not ≥ the active interval value so IMS forced the interval to the active value and issued asynchronous message DFS3812. Asynchronous message DFS3811 indicates the status of the change following the above action by IMS.

Example 11 for /CHANGE Command

The following commands illustrate how to change the maximum number of regions that can be simultaneously scheduled for a given transaction.

Entry ET:

```
/DISPLAY TRANSACTION SKS7
```

Response ET:

```
   TRAN   CLS ENQCT   QCT   LCT  PLCT CP NP LP SEGSZ SEGNO PARLM RC
SKS7       7     0     0 65535 65535  8  8  8     0     0     1  0
    PSBNAME: DFSDDLT7
    STATUS: BAL(  2)
*90226/134816*
```

Explanation: The status of transaction SKS7 indicates it is eligible for load balancing (BAL) and that two regions can be simultaneously scheduled.

Entry ET:

```
/CHANGE TRANSACTION SKS7 MAXRGN 4 (EOM)
```

Response ET:

```
DFS058I CHANGE COMMAND COMPLETED
```

Entry ET:

```
/DISPLAY TRANSACTION SKS7
```

Response ET:

```
   TRAN   CLS ENQCT   QCT   LCT  PLCT CP NP LP SEGSZ SEGNO PARLM RC
SKS7       7     0     0 65535 65535  8  8  8     0     0     1  0
    PSBNAME: DFSDDLT7
    STATUS: BAL(  4)
*90226/134845*
```

Explanation: The maximum number of regions that can be simultaneously scheduled for transaction SKS7 has been changed from 2 to 4.

Example 12 for /CHANGE Command

The following commands illustrate the changing of a unit of recovery.

Entry ET:

```
/CHANGE UOR 010040 ABORT
```

Response ET:

```
DFS058I CHANGE COMMAND COMPLETED
DFS0699I RESYNC ABORT  COMPLETE FOR PSB STLDDTL1, PRTKN=00010040,
       TOKEN IMS2    0000000100000000 IMS2
  *97226/134816*
```

Explanation: IMS backs out changes for pseudo recovery token 010040.

Entry ET:

```
/CHANGE UOR 010040 COMMIT
```

Response ET:

```
DFS058I CHANGE COMMAND COMPLETED
DFS0699I RESYNC COMMIT COMPLETE FOR PSB STLDDTL1, PRTKN=00010040,
       TOKEN IMS2    0000000100000000 IMS2
  *97226/134816*
```

Explanation: IMS makes changes for pseudo recovery token 010040 permanent.

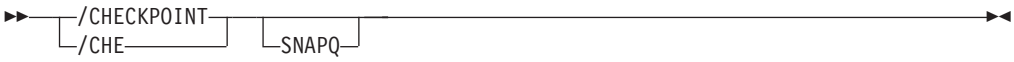
Chapter 9. /CHECKPOINT

Format

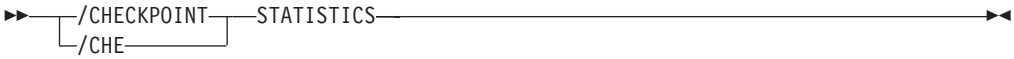
Shutdown Checkpoint

Attention: This command shuts down your IMS system. Be sure you understand the consequences of shutting down the system before you issue this command.

Simple Checkpoint



Statistics Checkpoint



Environments and Keywords

Table 25 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 25. Valid Environments for the /CHECKPOINT Command and Keywords

Command / Keyword	DB/DC	DBCTL	DCCTL
/CHECKPOINT	X	X	X
ABDUMP	X	X	X
DUMPQ	X		X
FREEZE	X	X	X
LEAVEPLEX	X	X	X
NOCQSSHUT	X		X
PURGE	X	X	X
QUIESCE	X		X
SNAPQ	X		X
STATISTICS	X	X	X

© Copyright IBM Corp. 1974, 2003

Usage

The /CHECKPOINT command records control and status information on the system log. /CHECKPOINT can be used to take a simple checkpoint of IMS, with the option of also dumping the contents of the message queue data sets to the system log or to shut down IMS normally. When IMS is shut down by the /CHECKPOINT command, it can be restarted with the /NRESTART command.

There are two conditions under which IMS cannot complete a shutdown normally. The most frequent is when there are multiple-page messages on a master terminal BTAM line, paging is not complete, and the master terminal is sharing a communication line with other physical terminals. It might be necessary to use the /ASSIGN command to reassign the master terminal to a line by itself so that the /IDLE command can be issued to terminate the multiple page output. If the master terminal cannot be reassigned to another line, assign it to the system console line.

The second condition occurs when a number of IMS-generated system messages are awaiting delivery to the master terminal. All system messages destined for the master terminal will be delivered, because they can impact the way the system is restarted. The master terminal operator should acknowledge delivery of a message by causing an I/O interrupt; that is, pressing PA2, which in turn causes another message to be sent, if one exists. Another option is to assign the master terminal to the system console.

When the /CHECKPOINT command is used to shut down IMS, the /BROADCAST command can be helpful in notifying the remote terminal operators that IMS is shutting down.

(blank)
Requests a simple checkpoint. Simple checkpoints are also invoked by IMS based on the number of entries to the system log. The number of log entries between simple checkpoints is specified during system definition. In an ETO environment, any dynamic nodes, LTERMs or users with no messages queued or status are deleted.

ABDUMP
Requests an abnormal termination dump of the IMS control region in addition to the shutdown option selected.

In a DBCTL environment, when /CHECKPOINT FREEZE ABDUMP is entered, all of the DBCTL address spaces are dumped.

FREEZE | DUMPQ | PURGE
Requests a shutdown of IMS. These variations of shutdown are provided for control over the method of stopping programs and lines, and to control the method of disposing of queues. The effects of these variations are shown in Table 26 on page 141 The request for a checkpoint shutdown might not be responded to immediately if any dependent regions are active. IMS will wait until these regions complete their current processing before continuing with the checkpoint shutdown. Message resynchronization for a specified transaction pipe does not affect IMS shutdown.

Restriction: A shutdown checkpoint is not allowed in a shared-queues environment if the CQS is not available.

Connections to external subsystems (not CCTL subsystems) will be quiesced. Connection attempts originating from dependent regions will be prohibited. After all dependent region connections have terminated, the control region will terminate its connection.

The DUMPQ and SNAPQ keywords designate starting points from which the message queue data sets can be rebuilt. However, the SNAPQ option dumps the message queues online while IMS is running without causing a shutdown of IMS.

The PURGE keyword attempts to empty all the queues so that no outstanding work remains. For DBCTL, IMS performs FREEZE processing because there are no message queues.

The FREEZE keyword shuts down IMS pending actions shown in Table 26 For DBCTL, active CCTL threads are allowed to complete before this keyword takes effect. This is also true when ABDUMP is used with this keyword. In the DBCTL environment, a `/CHECKPOINT` command with the PURGE keyword will be processed as though the FREEZE keyword was used instead of PURGE. `/CHECKPOINT FREEZE` in the DBCTL environment is correct because there are no message queues to empty.

In a DBCTL environment, when `/CHECKPOINT FREEZE ABDUMP` is entered, all of the DBCTL address spaces are dumped.

In a shared-queues environment, the `DUMPQ` and `PURGE` keywords cause IMS to shut down (as if you entered a `/CHECKPOINT FREEZE` command), but the message queues are not dumped or purged because the local IMS subsystem has no local queues. To dump the shared message queues when CQS terminates, use the `/CQSET` command before issuing the IMS shutdown checkpoint command.

Table 26 displays when to issue one of the three IMS shutdown options (FREEZE, DUMPQ, and PURGE) and the effect on the IMS resource.

Table 26. IMS Shutdown Options and Their Effect on IMS Resources

Resource Status	FREEZE Keyword	DUMPQ Keyword	PURGE Keyword
Message processing regions stopped	At program completion	At program completion	When transaction queues are empty
Batch message processing regions stopped	At checkpoint, SYNC call, or program completion	At checkpoint, SYNC call, or program completion	At program completion
Line input stopped	At message completion	At message completion	At message completion
Line output stopped	At message completion	At message completion	When all messages complete
Transaction and message queues	Retained in queue data sets	Dumped to system log	Emptied normally
Message-driven regions stopped	At message completion	At program completion	When transaction queues are empty
Fast Path output messages queued	Dumped to system log	Dumped to system log	Emptied normally
Fast Path input messages queued	Discarded	Emptied normally	Emptied normally
Fast Path DEDB online utility region	At program completion	At program completion	At program completion
Fast Path DEDBs	Closed	Closed	Closed
MSDBs	Dumped to MSDB checkpoint data set	Dumped to MSDB checkpoint data set	Dumped to MSDB checkpoint data set

Table 26. IMS Shutdown Options and Their Effect on IMS Resources (continued)

Resource Status	FREEZE Keyword	DUMPQ Keyword	PURGE Keyword
CCTL Regions	See *IMS Version 9: Application Programming: Database Manager* for a discussion of DRA threads.		

LEAVEGR
Deletes all affinities from the VTAM affinity table for the IMS subsystem on which this command is issued. Using this keyword removes the IMS subsystem from the generic resource group.

Recommendation: Cold start the DC component of an IMS subsystem that has been shut down with the LEAVEGR keyword to ensure that all affinities in IMS control blocks are also deleted. The IMS subsystem rejoins the generic resource group during startup.

If the VTAM ACB is closed (usually because of a /STOP DC command), a shutdown checkpoint command with the LEAVEGR keyword is rejected.

LEAVEPLEX
In an IMSplex, this keyword is specified if the IMS that is being shut down is not going to rejoin the IMSplex. Specify the LEAVEPLEX keyword when you do not intend to bring the IMS back up in the IMSplex.

If LEAVEPLEX is specified, and the IMS is a member of an IMSplex with global online change enabled, an attempt is made to remove the IMS's ID from the OLCSTAT data set. If there is any error in removing the IMS ID from the OLCSTAT, message, DFS3443, DFS3444, or DFS3448 is written out to the system console and the IMS is shut down. In this case, the IMS ID may still be in the OLCSTAT data set and will have to be deleted using the DFSUOLC utility.

NOCQSSHUT
Is used with the /CHE DUMPQ, /CHE FREEZE, or /CHE PURGE commands to **not** shut down the CQS address space when the IMS control region terminates. The CQS address space remains active and connected to the message queue structures. NOCQSSHUT is only applicable when IMS is running in a shared-queues environment. The default is to shut down the CQS address space when the IMS control region terminates.

QUIESCE
Halts processing of all VTAM terminals. When QUIESCE is specified, IMS sends the VTAM shutdown indicator to all VTAM terminals and waits until these nodes have completed processing before performing the normal checkpoint shutdown. During the processing of a quiesce shutdown, the master terminal operator might want to terminate the VTAM network without waiting for the orderly termination to complete. This can be done by entering the /CHECKPOINT command again, either with FREEZE, DUMPQ, or PURGE but without QUIESCE.

SNAPQ
Requests a simple checkpoint and dumps the contents of the message queues to the system log.

In a shared-queues environment, /CHECKPOINT SNAPQ does not snap the queues because the local IMS subsystem has no local queues. Use the /CQCHKPT command to initiate a CQS structure checkpoint.

In an XRF environment, /CHECKPOINT SNAPQ synchronizes the active and alternate IMS subsystems.

STATISTICS
Requests that IMS performance records be created and written to the system log. No other checkpoint processing occurs.

The /CHECKPOINT STATISTICS command does not create a system checkpoint on the log.

Examples

Example 1 for /CHECKPOINT Command

Entry ET:

```
/CHECKPOINT
```

Response ET:

```
DFS058I  (time stamp) CHECKPOINT COMMAND IN PROGRESS
DFS994I *CHKPT 82102/110247**SIMPLE*
```

Explanation: A simple checkpoint of IMS is written to the system log at 110247 (time) on 82102 (Julian date). The checkpoint number is 82102/110247.

Example 2 for /CHECKPOINT Command

Entry ET:

```
/CHECKPOINT FREEZE
```

Response ET:

```
DFS058I  (time stamp) CHECKPOINT COMMAND IN PROGRESS
DFS994I *CHKPT 82206/120118**FREEZE*
```

Explanation: IMS is terminated after all checkpoint freeze functions complete. The checkpoint is written to the system log at 120118 (time) on 82206 (Julian date). The checkpoint number is 82206/120118.

Example 3 for /CHECKPOINT Command

Entry ET:

```
/CHECKPOINT FREEZE QUIESCE
```

Response ET:

```
DFS058I  (time stamp) CHECKPOINT COMMAND IN PROGRESS
DFS994I *CHKPT 82102/110247**FREEZE*
```

Explanation: IMS is terminated after all VTAM nodes have returned a shutdown-complete indicator to IMS and IMS has completed all checkpoint freeze functions. The checkpoint is written to the system log at 110247 (time) on 82102 (Julian date). The checkpoint number is 82102/110247.

Example 4 for /CHECKPOINT Command

Entry ET:

```
/CHECKPOINT PURGE ABDUMP
```

Response ET:

```
DFS058I  (time stamp) CHECKPOINT COMMAND IN PROGRESS
DFS994I *CHKPT 82128/101112**PURGE*
```

Explanation: IMS is terminated after all checkpoint purge functions complete. The checkpoint is written to the system log at 101112 (time) on 82128 (Julian date). The checkpoint number is 82128/101112.

An z/OS ABEND message is issued when the dump of the IMS control region completes.

Example 5 for /CHECKPOINT Command

Entry ET:

```
/CHECKPOINT PURGE
```

Response ET:

```
DFS058I  (time stamp) CHECKPOINT COMMAND IN PROGRESS
DFS994I *CHKPT 82103/131415**PURGE*
```

Explanation: IMS is terminated after all checkpoint purge functions complete. The checkpoint is written to the system log at 131415 (time) on 82103 (Julian date). The checkpoint number is 82103/131415.

Example 6 for /CHECKPOINT Command

Entry ET:

```
/CHECKPOINT STATISTICS
```

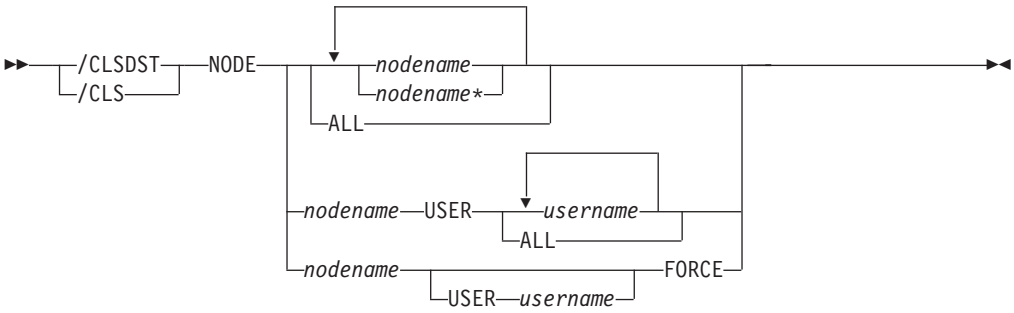
Response ET:

```
/DFS058I  (timestamp) CHECKPOINT COMMAND IN PROGRESS
/DFS994I  STATISTICS CHECKPOINT COMMAND COMPLETED
```

Explanation: IMS performance statistics records are written to the system log. No other information is saved. This checkpoint cannot be used for a system restart.

Chapter 10. /CLSDST

Format



Environments and Keywords

Table 27 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 27. Valid Environments for the /CLSDST Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/CLSDST	X		X
FORCE	X		X
NODE	X		X
USER	X		X

Usage

/CLSDST causes IMS to disconnect a VTAM terminal.

/CLSDST resets preset mode, test mode, response mode, lock node, lock LTERM, pstop LTERM, and purge LTERM, because these statuses are not significant and therefore are not kept after a logon or restart. /CLSDST will also do some other cleanup depending on the recovery settings for the node. Below are the actions taken:

RCVYSTSN=NO

/CLSDST acts like a /CHANGE NODE COLDSESS command for FINANCE and SLUP nodes by setting the session status to “cold”. /CLSDST will act like a /QUIESCE NODE command for ISC (LU6.1) nodes by initiating the shutdown and deallocating the user for the specified node. This action changes the session status to cold. With these actions taken by the /CLSDST command, the next session initiation request for this node will be allowed to again attempt a session cold start. For ETO nodes, the control block structure could be deleted, if no significant status exists.

RCVYCONV=NO

/CLSDST causes any IMS conversations (active and held) to be terminated. Any conversational message that is queued or being processed will have its output response message delivered asynchronously.

© Copyright IBM Corp. 1974, 2003

RCVYFP=NO
/CLSDST causes Fast Path status and messages to be discarded.

If global resource information is not kept in Resource Manager (RM), /CLSDST logs a node off and resets status locally. If global resource information is kept in RM, /CLSDST resets status globally. If the node has no significant status, /CLSDST deletes the node in RM.

If ROUTE is specified, it should be specified with ROUTE(*). The command fails if not routed to the IMS where the node is active.

FORCE
Allows IMS to reinitialize a hung node with I/O in progress. FORCE reinitializes one session and is valid only on an active IMS system. To activate FORCE, use the command /CLSDST NODE P1 FORCE. For ISC nodes with parallel sessions, the NODE and USER keywords must be specified and only one USER parameter is allowed.

FORCE operates under the following conditions:

- A display from VTAM indicates that no session exists. If a session does exist, or is in process, you must issue the VTAM command VARY NET,INACT,FORCE to terminate the session, and the command must successfully complete. This command terminates all parallel sessions.

  Attention: If VARY NET,INACT,FORCE is not issued or does not successfully complete before you issue the FORCE command, and the terminal is in the process of creation or termination, the results might be unpredictable.
- The session is connected to IMS and output is in progress. A /DISPLAY on the node indicates that a CID exists, that the node is connected (a status of CON is displayed), and that the node is not idle (a status of IDLE is not displayed).

NODE
Specifies the node to be disconnected by IMS. The specified node must be connected before a command is issued to disconnect it, as indicated by CON on the /DISPLAY NODE command referring to that terminal.

If the USER keyword is omitted for ISC nodes, all half-sessions of a session type 6 node are terminated. If the USER keyword is omitted, generic parameters are allowed for the NODE keyword.

The timing of the disconnection depends on the type of terminal:

- For keyboards or printers, console components and interactive terminals, the disconnection occurs at the next message boundary.
- For component types that group messages (such as a SLU 1 statement reader, printer, or disk), the disconnection occurs at the end of any group where processing is in progress.
- For 3270 displays, the disconnection occurs at the completion of the current (if any) I/O operation.

USER
Must be specified with the NODE keyword. NODE USER specifies the ISC user allocated to the ISC node or the dynamic user signed on to the dynamic node. When a /CLSDST NODE nodename USER username command is issued, it only affects the NODE if the USER is still associated with the node.

For non-ISC dynamic nodes, this command is valid only if the user is still signed on to the node. For ISC nodes, the half-sessions of the ISC node

allocated to the specific users are terminated and the users are not deallocated from the session. On restart, the /OPNDST command must specify the same users and ID pairs.

Example for /CLSDST Command

Entry ET:

```
/CLSDST NODE WEST
```

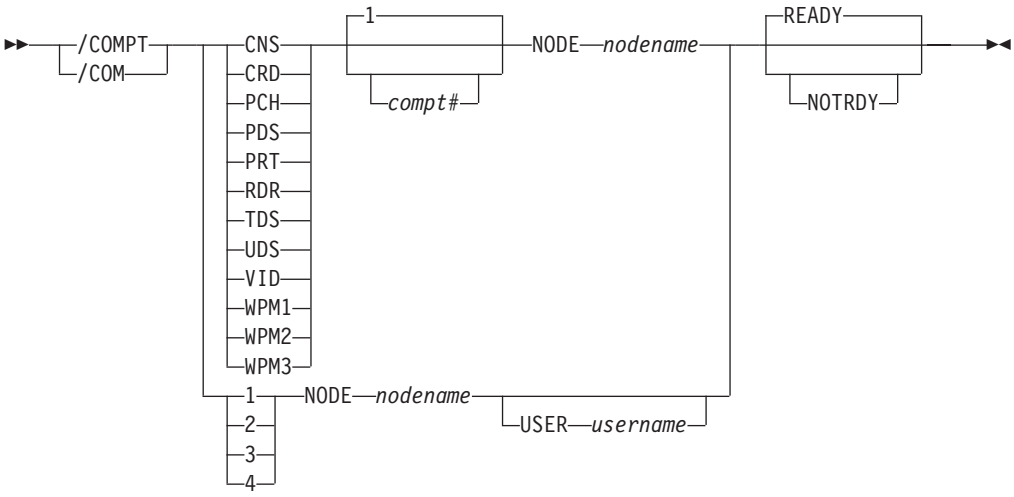
Response ET:

```
DFS058I  CLSDST COMMAND COMPLETED
```

Explanation: The node, WEST, is disconnected from IMS.

Chapter 11. /COMPT

Format



Environments and Keywords

Table 28 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 28. Valid Environments for the /COMPT Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/COMPT	X		X
CNS	X		X
CRD	X		X
NODE	X		X
NOTRDY	X		X
PCH	X		X
PDS	X		X
PRT	X		X
RDR	X		X
READY	X		X
TDS	X		X
UDS	X		X
USER	X		X
VID	X		X
WPM1	X		X
WPM2	X		X
WPM3	X		X

© Copyright IBM Corp. 1974, 2003

Usage

/COMPT sets a particular terminal component to a ready/not ready state. Output messages queued for a particular component will not be sent unless the component is ready. Depending on terminal type and the availability of messages queued for other components, output operations for other components can continue.

The ready/not ready state set by the /COMPT command can be altered by the following:

- Another /COMPT command
- A /START, /RSTART, or /RCOMPT command
- An I/O error on the terminal component

/COMPT can only refer to a VTAM-attached terminal component.

Restriction: When the /COMPT command contains the keyword CRD, it cannot also contain any of the following keywords: WPM1, WPM2, or WPM3.

The command format takes two forms. That is, a component can be referenced by using a keyword, such as VID, or by using a number, such as 2.

When a keyword is used, a search is made of the components (as defined in the TERMINAL macro during IMS system definition) for the component type defined that corresponds to the specified keyword.

When a match is found, that component is made ready/not ready as specified by the command. If a number other than 1 follows the keyword, the corresponding occurrence of that component type is made ready/not ready.

/COMPT supports up to 4 components. When a number from 1 through 4 is used instead of a keyword, the component affected is the one defined in that position during system definition, independent of component type. The USER keyword is required when nodename is an ISC node with users.

For more information about component support, see Chapter 5, “/ASSIGN,” on page 99

Examples

Example 1 for /COMPT Command

Entry ET:

```
/COMPT 4 NODE ABC READY
```

Response ET:

```
DFS058I  COMPT COMMAND COMPLETED
```

Explanation: The fourth component defined in the TERMINAL macro on node ABC is made ready to IMS.

Example 2 for /COMPT Command

Entry ET:

```
/COMPT VID 2 NODE ABC READY
```

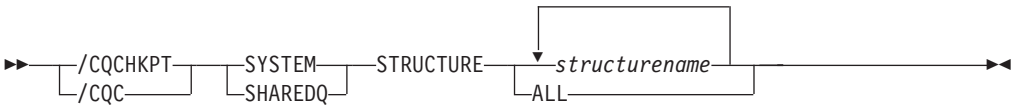
Response ET:

```
DFS058I  COMPT COMMAND COMPLETED
```

Explanation: The second display component on node ABC is declared operable to IMS.

Chapter 12. /CQCHKPT

Format



Environments and Keywords

Table 29 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 29. Valid Environments for the /CQCHKPT Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/CQCHKPT	X		X
SHAREDQ	X		X
STRUCTURE	X		X
SYSTEM	X		X

Usage

/CQCHKPT initiates a CQS checkpoint for a specific coupling facility list structure or all the coupling facility list structures to which the IMS subsystem is connected.

This command sends the DFS058 CQCHKPT COMMAND IN PROGRESS message to the inputting terminal, and sends an asynchronous response to the system console and master terminal when the CQS checkpoint is complete.

This command is valid only in a shared-queues environment.

SHAREDQ
Specifies that the entire queue structure is to be checkpointed to the structure recovery data set. While the checkpoint is in progress for the structure, no CQS can access the structure.

During a structure checkpoint, every CQS connected to that structure also takes a system checkpoint.

Recommendation: When possible, issue this command when it will have the least performance impact to your online IMS subsystems.

STRUCTURE
Specifies a specific structure name (or all) for which a CQS checkpoint is to be taken. If an overflow structure exists for a structure, a checkpoint is taken for both the primary and overflow structure.

SYSTEM
Specifies that a system checkpoint is to be taken; the CQS internal tables are checkpointed and written to the CQS log. Only the CQS for which you enter the command takes a system checkpoint.

© Copyright IBM Corp. 1974, 2003

Examples

Example 1 for /CQCHKPT Command

Entry ET:

```
/CQCHKPT SYSTEM STRUCTURE IMSMSGQ01
```

Response ET:

```
DFS058I CQCHKPT COMMAND IN PROGRESS
CQS0030I SYSTEM CHECKPOINT COMPLETE, STRUCTURE IMSMSGQ01       ,
 LOGTOKEN 0000000001687D3F CQS1CQS
DFS1972I CQCHKPT SYSTEM  COMMAND COMPLETE FOR STRUCTURE=IMSMSGQ01
```

Explanation: A CQS system checkpoint completes successfully.

Recommendation: Record the log token displayed in the CQS0030I message because you might need it for a CQS restart.

Example 2 for /CQCHKPT Command

Entry ET:

```
/CQCHKPT SHAREDQ STRUCTURE IMSMSGQ01
```

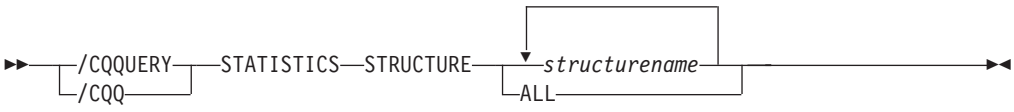
Response ET:

```
DFS058I CQCHKPT COMMAND IN PROGRESS
CQS0220I CQS CQS1CQS  STARTED STRUCTURE CHECKPOINT FOR
 STRUCTURE IMSMSGQ01        CQS1CQS
CQS0200I STRUCTURE IMSMSGQ01        QUIESCED FOR
 STRUCTURE CHECKPOINT CQS1CQS
CQS0201I STRUCTURE IMSMSGQ01        RESUMED AFTER
 STRUCTURE CHECKPOINT CQS1CQS
CQS0030I SYSTEM CHECKPOINT COMPLETE, STRUCTURE IMSMSGQ01       ,
 LOGTOKEN 0000000001688652 CQS1CQS
CQS0221I CQS CQS1CQS  COMPLETED STRUCTURE CHECKPOINT
 FOR STRUCTURE IMSMSGQ01        CQS1CQS
DFS1972I CQCHKPT SHAREDQ COMMAND COMPLETE FOR STRUCTURE=IMSMSGQ01
```

Explanation: A CQS structure checkpoint for a specific structure completes successfully.

Chapter 13. /CQQUERY

Format



Environments and Keywords

Table 30 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 30. Valid Environments for the /CQQUERY Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/CQQUERY	X		X
STATISTICS	X		X
STRUCTURE	X		X

Usage

The /CQQUERY command displays information regarding a specific coupling facility list structure or all the coupling facility list structures holding IMS messages. The /CQQUERY STATISTICS command displays the statistics and status information for the coupling facility list structures specified by the STRUCTURE keyword. The following statistics information is displayed:

- Number of data elements that can be allocated in the structure
- Number of list entries that can be allocated in the structure
- Number of data elements in use in the structure
- Number of list entries in use in the structure
- Entry to element ratio

This command is valid only in a shared-queues environment.

In an IMSplex, /CQQUERY displays information regarding a specific shared queue coupling facility list structure or all the shared queue coupling facility list structures holding IMS messages. /CQQUERY does not display any information about resource structures.

When the /CQQ command is issued through OM, command processing is not serialized through the IMS CTL TCB.

STATISTICS
Specifies that statistics should be gathered and displayed.

STRUCTURE
Specifies that one or more structure names follow.

Only primary, coupling facility list-structure names used by IMS for shared queues are valid.

© Copyright IBM Corp. 1974, 2003

Examples

Example 1 for /CQQUERY Command

Entry ET:

```
/CQQUERY STATISTICS STRUCTURE ALL
```

Response ET:

```
 STRUCTURE NAME    LEALLOC  LEINUSE ELMALLOC ELMINUSE    LE/EL
 IMSMSGQ01             272        4      541        4  0001/0002
 IMSMSGQ01OFLW         N/A      N/A      N/A      N/A     N/A
 IMSEMHQ01             272        3      541        3  0001/0002
 IMSEMHQ01OFLW         N/A      N/A      N/A      N/A     N/A
 *97211/120123*
```

Explanation: This command displays the structure statistics of the message queue and Fast Path EMH queue structures used by IMS. The list entries allocated (LEALLOC), the list entries in use (LEINUSE), the elements allocated (ELMALLOC) and the elements in use (ELMINUSE) show the usage of the structures. When CQS allocates a structure, it allocates a certain number of list entries and elements to manage data on the structure. A structure is full if all list entries are in use or if all elements are in use.

Example 2 for /CQQUERY Command

Entry ET:

```
/CQS STATISTICS STRUCTURE IMSMSGQ01
```

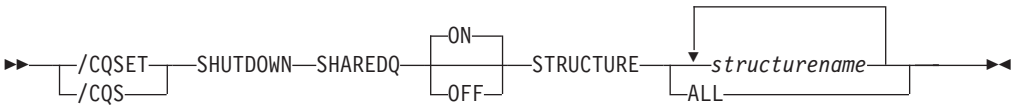
Response ET:

```
STRUCTURE NAME    LEALLOC  LEINUSE ELMALLOC ELMINUSE   LE/EL
IMSMSG101            9132     9027    9130       9071 0001/0001
IMSMSGQ01OFLW        1915     1866    1912       1866 0001/0001
*00039/180909*
```

Explanation: This command displays the structure statistics of the message queue structure used by IMS and its associated overflow queue.

Chapter 14. /CQSET

Format



Environments and Keywords

Table 31 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 31. Valid Environments for the /CQSET Command and Keywords

Command / Keyword	DB/DC	DBCTL	DCCTL
/CQSET	X		X
SHAREDQ	X		X
SHUTDOWN	X		X
STRUCTURE	X		X

Usage

Use a `/CQSET SHUTDOWN SHAREDQ ON|OFF` command to tell CQS whether to take a structure checkpoint during normal shutdown. You can specify a structure checkpoint for a specific coupling facility or for all coupling facility list structures used by IMS.

IMS initiates a normal CQS shutdown during a normal IMS shutdown.

This command is valid only in a shared-queues environment.

In an IMSplex, when the `/CQS` command is issued through OM command processing, it is not serialized through the IMS CTL TCB.

SHAREDQ
Specifies that the entire queue structure is to be checkpointed to the structure recovery data set. While the checkpoint is in progress for the structure, no CQS can access the structure.

During a structure checkpoint, every CQS connected to that structure also takes a system checkpoint.

SHUTDOWN
Specifies that CQS should take a structure checkpoint during normal CQS shutdown.

STRUCTURE
Specifies a specific structure name (or all) for which a CQS checkpoint is to be taken. If an overflow structure exists for a structure, a checkpoint is taken for both the primary and overflow structure.

© Copyright IBM Corp. 1974, 2003

Example for /CQSET Command

Entry ET:

```
/CQSET SHUTDOWN SHAREDQ ON STRUCTURE ALL
```

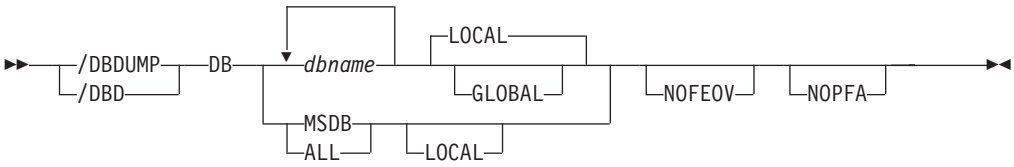
Response ET:

```
DFS058I  CQSET COMMAND COMPLETE
```

Explanation: CQS will take a structure checkpoint when it shuts down.

Chapter 15. /DBDUMP

Format



Environments and Keywords

Table 32 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 32. Valid Environments for the /DBDUMP Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/DBDUMP	X	X	
DB	X	X	
GLOBAL	X	X	
LOCAL	X	X	
NOFEOV	X	X	
NOPFA	X	X	

Usage

/DBDUMP is used to prevent transactions or programs from updating DL/I databases. It also can be used to dump all MSDBs to the MSDB dump data set. /DBDUMP does not apply to DEDBs.

The /DBDUMP command can be used on HALDBs. For more information see Appendix H, “High Availability Large Database Commands,” on page 867

For the results of issuing this command on a shared secondary index, see Appendix D, “Shared Secondary Index Database Commands,” on page 849

In an IMSplex, the output of the /DBD command is changed when the command is entered through the OM API. In this case, the DFS058I message is not returned to OM. The command response returned to OM contains one or more of the following messages as appropriate to the database type and the command completion.

Full Function Database messages: DFS132, DFS160, DFS216, DFS0488I, DFS1407, DFS2026, DFS3318I, DFS3320I, DFS3321I, DFS3325I, DFS3462I, DFS3463I, DFS3466I

DB

Specifies the databases to which the /DBDUMP command applies. When the /DBDUMP command is entered, the message processing regions using the

© Copyright IBM Corp. 1974, 2003

specified databases are terminated at the conclusion of processing their current transactions, in preparation to close the database and allow it to be opened for input only.

If a DL/I database specified in the command is being used by a batch message processing region, an error message is returned to the master terminal. When this message is issued, the command is ignored for the database named in the message; processing continues for the other databases specified in the command. The master terminal operator must wait until the batch message processing concludes processing before reentering the command.

As the message processing regions terminate programs, the data sets of the named databases in the command are closed. The IMS log switches to the next OLDS. This switch to the next OLDS is marked as a recovery point for log archiving purposes. IMS issues a simple checkpoint. The scheduling of transactions is then resumed, although no transactions will be allowed to update the specified databases. Programs with update intent will be scheduled, but update calls to the database will result in a 3303 pseudoabend or a BA status if the INIT call was issued.

/DBDUMP can be used to dump all the MSDBs to the MSDB dump data set by specifying the reserved parameter MSDB with the DB keyword when entering the /DBDUMP DB command or by entering the /DBDUMP DB ALL command. The MSDBs dumped to the MSDB dump data set can be used as input to the MSDB dump recovery utility. A specific MSDB cannot be a parameter of the DB keyword.

The /START DB command resets the effect of the /DBDUMP command. The /START command is not required for MSDBs, because the data for these databases resides in processor storage, and the databases are never closed.

For DBCTL, when CCTL schedules a PSB, the DBCTL thread SCHED request defines the thread as LONG or SHORT. If the database is currently scheduled to a LONG thread, the command is rejected; otherwise, the thread is allowed to “complete” before the database is acted upon. This results in either a commit point or transaction termination.

GLOBAL

Applies when an IRLM is active and specifies that the command applies to all online subsystems sharing the database. The /DBDUMP command with the GLOBAL keyword puts the database in read status and prevents transactions from updating the database in all online subsystems that share the database.

The GLOBAL keyword is mutually exclusive with the ALL parameter or the MSDB parameter and causes the command to be rejected if both parameters are specified. The GLOBAL keyword requires that IRLM be active and will cause the command to be rejected if IRLM is not active.

If the GLOBAL keyword on a command is entered from an OM API, the command should only be routed to one IMS system in the IMSplex. The IMS that receives the command from OM will make DBRC calls to update the RECON with GLOBAL status. It will also request IRLM NOTIFY to route and process the command on sharing IMS systems, and then process the command locally.

Messages produced on the NOTIFIED systems will appear only on the system console and will not be routed back to the OM API which originally entered the command.

If multiple IMS systems have been explicitly specified in the route list, the master IMS system will process the command as described above. However,

the non-master IMS systems, to which OM routes the command, will reject the command with the return and reason codes shown in Table 33

Table 33. Return and reason code for the GLOBAL keyword issued from the OM API

Return Code	Reason Code	Meaning
X'00000004'	X'00001000'	The command contained the GLOBAL keyword and was routed to more than one IMS system in the IMSPLEX. The non-master IMS systems will reject this command when OM routes the command to them. The master IMS system will process this command and use IRLM NOTIFY to route and process the command on the non-master IMS systems. See the discussion under the GLOBAL keyword.

LOCAL
Specifies that the command only applies to the subsystem in which the command is entered. This command does not affect any other subsystem sharing the database. The LOCAL keyword can be used to restrict concurrent updates. LOCAL is the default.

NOFEOV
Specifies that there is no forced end of volume, so that the IMS log does not switch to the next OLDS. If NOFEOV is specified without the MSDB keyword, a simple checkpoint is not taken.

NOPFA
Specifies that DBRC is not notified that the database or area has changed status. You can use this keyword when you need to authorize the database for use after it is offline, for example, for offline utilities. By using this keyword, DBRC does not prevent further authorizations for the database or area.

Recommendation: Before starting the database or area, reissue this command **without** the NOPFA keyword to inform DBRC of the change in status for the database or area.

Examples

Example 1 for /DBDUMP Command

Entry ET:

```
/DBDUMP DATABASE PAYROLL
```

Response ET:

```
DFS058I  (time stamp) DBDUMP COMMAND IN PROGRESS
```

Explanation: Currently executing application programs are being terminated. When the termination completes, the databases are stopped for update and the output log is switched to the next OLDS.

Response ET:

```
DFS0488I DBD COMMAND COMPLETED.
         DBN=PAYROLL RC=0
DFS3257I ONLINE LOG NOW SWITCHED FROM DFSOLP( ) TO DFSOLP( )
DFS994I *CHKPT 82080/111213**SIMPLE*
```

Explanation: The new OLDS is used to record a simple checkpoint at 111213 (time) on 82080 (Julian date). The checkpoint number is 82080/111213. All /DBDUMP command functions are complete. The /START DATABASE command must be used to start the database after the dump job completes.

Example 2 for /DBDUMP Command

Entry ET:

```
/DBDUMP DATABASE MSDB
```

Response ET:

```
DFS058I (time stamp) DBDUMP COMMAND IN PROGRESS
```

Explanation: All MSDBs are dumped to the MSDB dump data set because MSDB was specified as the parameter of the database keyword.

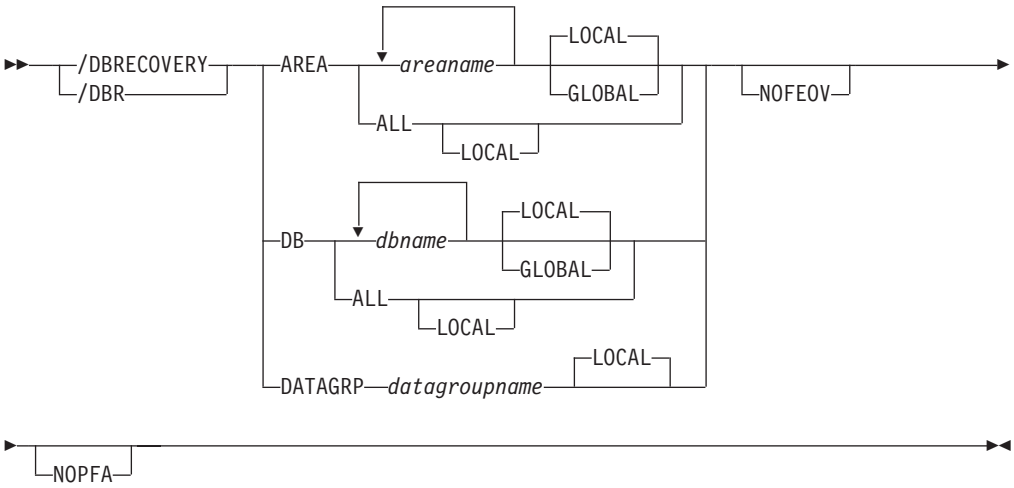
Response ET:

```
DFS994I CHKPT 82069/123624**SIMPLE*
```

Explanation: A simple checkpoint is recorded on the new system log at 123624 (time) on 82069 (Julian date). The checkpoint number is 82069/123624. All MSDBs are dumped.

Chapter 16. /DBRECOVERY

Format



Environments and Keywords

Table 34 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 34. Valid Environments for the /DBRECOVERY Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/DBRECOVERY	X	X	
AREA	X	X	
DB	X	X	
DATAGRP	X	X	
GLOBAL	X	X	
LOCAL	X	X	
NOFEOV	X	X	
NOPFA	X	X	

Usage

/DBRECOVERY is used to prevent transactions or programs from accessing DL/I databases, DEDBs, or DEDB areas.

The /DBRECOVERY command can be used on HALDBs. For more information see Appendix H, “High Availability Large Database Commands,” on page 867

Use of the ALL parameter indicates that the command applies to all databases, even those not opened.

© Copyright IBM Corp. 1974, 2003

/DBRECOVERY closes and deallocates the databases, and unauthorizes them with DBRC. Once the database or area referenced by /DBRECOVERY is closed, the IMS log switches to the next OLDS (unless you specify the NOFEOV keyword). This switch to the next OLDS is marked as a recovery point for log archiving purposes. IMS issues a simple checkpoint.

/DBRECOVERY is supported on an RSR tracking subsystem, but only for a database readiness level (DLT) subsystem. /DBRECOVERY is used on an RSR tracking subsystem to take shadow areas and databases offline for image copy and recovery. The /DBRECOVERY command also stops online forward recovery (OFR) if it is in progress for the specified database or area.

IMS databases, DEDBs, and DEDB areas closed by the master terminal operator with the /DBRECOVERY command can be used as input to the offline IMS Database Recovery utility. The Database Recovery utility runs in a batch region.

For the results of issuing this command on a shared secondary index, see Appendix D, “Shared Secondary Index Database Commands,” on page 849

The /DBRECOVERY command will not deallocate a data set if a VSAM data set hardware error occurred. Use VARY nnn,OFFLINE,FORCE to deallocate the data set (where nnn is the is the number of the device). For information on recovering VSAM data sets, see *DFSMS/MVS Managing Catalogs* (SC26-4914).

In an IMSplex, the output of the /DBR command is different when the command is entered through the OM API. In this case, the DFS058I message is not returned to OM. The command response returned to OM contains one or more of the following messages as appropriate to the database type and the command completion. For more information about specific messages, see *IMS Version 9: Messages and Codes, Volume 1*.

- For /DBR AREA commands that specify GLOBAL, only the command master returns the asynchronous messages to OM. When a command is processed with the LOCAL keyword, all IMSs are able to return the asynchronous messages to OM.
- Full Function Database messages for /DBR DB: DFS132, DFS160, DFS216, DFS0488I, DFS0565I, DFS1407, DFS2026, DFS3318I, DFS3320I, DFS3321I, DFS3464I, DFS3466I.
- Fast Path Database messages for /DBR DB: DFS140I, DFS0666I, DFS3062I, DFS3320I.
- Fast Path Database messages for /DBR AREA: DFS140I, DFS0488I, DFS0666I, DFS1407I, DFS3060I, DFS3062I, DFS3320I, DFS3325I, DFS3342I, DFS3720I.

AREA

Specifies the DEDB areas (but not DEDBs) to be processed by the /DBRECOVERY command. /DBRECOVERY AREA stops the accessing and updating of specified DEDB areas and closes them.

/START AREA can be used to reopen and reallocate the areas closed by /DBRECOVERY AREA command.

The /DBRECOVERY command for Fast Path Virtual Storage Option (VSO) areas in a z/OS data space removes the areas from the data space and forces updates to be written back to DASD.

DB

Specifies the DL/I databases and Fast Path DEDBs (and inclusive areas) to be processed by the /DBRECOVERY command. MSDBs cannot be specified with

3 /DBRECOVERY. /DBRECOVERY DB allows scheduling of transactions or programs that
3 use the database to continue after successful completion of the command.
3 However, calls to DEDB databases will receive an FH status code. If the
3 application program is prepared to accept status codes regarding data
3 unavailability through the INIT STATUS GROUPA or GROUPB call, calls to DL/I
3 databases will result in either a 3303 pseudoabend, a BA, or BB status code.

3 When /DBRECOVERY DB is entered, the data sets of the databases named in the
3 command are closed. The /DBRECOVERY DB command dynamically deallocates
3 the IMS databases. For DEDBs, the randomizer is unloaded and removed from
3 storage.

3 If a DL/I database is in use by a batch message processing region when the
3 /DBRECOVERY command is issued, a DFS0565I message is returned to the master
3 terminal, the command is ignored for the database named in the message, and
3 processing continues for any other databases specified in the command. The
3 master terminal operator must wait until the BMP ends before reissuing the
3 /DBRECOVERY command to close the databases named in any DFS0565I
3 messages. DEDB databases will accept the /DBRECOVERY command while
3 actively being used by a batch message processing region. However, all further
3 calls to the database will receive an FH status code.

2 The region identified in the DFS0565I message can also be an MPP region. The
2 error message can be issued in order to break a potential deadlock condition
2 between Online Change, a /DBR, or /STA command against a database being
2 used by the MPP and an application running in the MPP issuing an ICMD or
2 CMD call.

3 The /START DB command is used to reallocate the databases closed by the
3 /DBRECOVERY DB command except for HALD DB partitions. A HALDB partition
3 can be reallocated if the partition has either EEQEs, was previously authorized
3 but not allocated, or the OPEN keyword was used. HALDB partitions are
3 allocated at first reference. For DEDBs, a /START DB command also causes a
3 reload of the randomizer.

3 For DBCTL, when CCTL schedules a PSB, the DBCTL thread SCHED request
3 defines the thread as LONG or SHORT. If the database is currently scheduled
3 to a LONG thread, the command is rejected; otherwise, the thread is allowed to
3 “complete” before the database is acted upon. This results in either a commit
3 point or transaction termination.

3 On an RSR tracking subsystem, /DBR DB for a DEDB is not permitted.

DATAGRP

Specifies a **group** of DL/I databases and Fast Path DEDBs to be processed by the /DBRECOVERY command. Use the INIT.DBDSGRP command with the GRPNAME and DBGRP keywords to define the data group in the RECON data set.

DATAGRP is valid on active and RSR tracking subsystems.

Related Reading: See “Group Parameters” on page 20 for more information on defining data groups.

Recommendation: Although you can use DBDS groups as well as database groups for this command, you should use database groups whenever possible to eliminate the overhead of converting the DBDS group to a database group.

GLOBAL

Applies when the database is registered to DBRC and specifies that the command applies to all sharing online subsystems. /DBRECOVERY GLOBAL can be

used to close the database so that it can be taken offline when this command successfully completes in all online subsystems. DBRC will prevent further authorization.

The GLOBAL keyword and the ALL parameter are mutually exclusive. The /DBRECOVERY command will be rejected if both GLOBAL and ALL are specified. The IRLM must be active when the GLOBAL keyword is used. If IRLM is not active, the command is rejected.

If the GLOBAL keyword on a command is entered from an OM API, the command should only be routed to one IMS system in the IMSplex. The IMS that receives the command from OM will make DBRC calls to update the RECON with GLOBAL status. It will also request IRLM NOTIFY to route and process the command on sharing IMS systems, and then process the command locally.

Messages produced on the NOTIFIED systems will appear only on the system console and will not be routed back to the OM API which originally entered the command.

If multiple IMS systems have been explicitly specified in the route list, the master IMS system will process the command as described above. However, the non-master IMS systems, to which OM routes the command, will reject the command with the return and reason codes in Table 35

Table 35. Return and reason code for GLOBAL keyword issued from the OM API

Return Code	Reason Code	Meaning
X'00000004'	X'00001000'	The command contained the GLOBAL keyword and was routed to more than one IMS system in the IMSPLEX. The non-master IMS systems will reject this command when OM routes the command to them. The master IMS system will process this command and use IRLM NOTIFY to route and process the command on the non-master IMS systems. See the discussion under the GLOBAL keyword.

Restriction: When the GLOBAL keyword is specified, the command is performed for the named database only. If the command is to be performed for a HIDAM database and its index, both must be named.

The GLOBAL keyword is not supported for an RSR tracking subsystem. The GLOBAL keyword cannot be combined with the DATAGROUP keyword.

LOCAL
Specifies that the command only applies to the subsystem in which the command is entered. This command does not affect any other subsystem sharing the database or area. Use the LOCAL keyword to increase usability for the other system sharing the database or area. LOCAL is the default.

NOFEOV
Specifies that there is no forced end of volume, so that the log does not switch to the next OLDS and simple checkpoint is not taken. The NOFEOV keyword is the default for an RSR tracking subsystem so that the OLDS is not switched, nor is a simple checkpoint taken.

NOPFA
Specifies that DBRC is not notified that the database or area has changed status. You can use this keyword when you need to authorize the database for

use after it is offline, for example, for offline utilities. By using this keyword, DBRC does not prevent further authorizations for the database or area.

Recommendation: Before starting the database or area, reissue this command **without** the NOPFA keyword to inform DBRC of the change in status for the database or area.

Examples

Example 1 for /DBRECOVERY Command

Entry ET:

```
/DBRECOVERY AREA ALL
```

Response ET:

```
DFS058I  DBRECOVERY COMMAND IN PROGRESS
DFS0488I  DBRECOVERY COMMAND COMPLETED. AREA=DBAREA0
DFS0488I  DBRECOVERY COMMAND COMPLETED. AREA=DBAREA1
```

Explanation: DEDB areas are being stopped for inquiry and update, closed, and deallocated; the output log is switched to the next OLDS.

Example 2 for /DBRECOVERY Command

Entry ET:

```
/DBRECOVERY DATABASE SKILL
```

Response ET:

```
DFS058I  DBRECOVERY COMMAND IN PROGRESS
DFS2500I DATABASE SKILL SUCCESSFULLY DEALLOCATED
DFS0488I DBRECOVERY COMMAND COMPLETED. DBN=SKILL RC=0
DFS3257I ONLINE LOG NOW SWITCHED FROM DFSOLP( ) TO DFSOLP( )
DFS944I *CHKPT 82123/121314**SIMPLE**
```

Explanation: The new OLDS is used to record a simple checkpoint at 121314 (time) on 82123 (Julian date). The checkpoint number is 82123/121314. All /DBRECOVERY command functions are complete. The Database Recovery utility can be executed after archiving. Currently executing application programs are being terminated. When the termination completes, the SKILL database is stopped for inquiry and update, closed, and deallocated; the output log is switched to the next OLDS. /START DATABASE must be used to start the database after the recovery job completes.

Example 3 for /DBRECOVERY Command

This example shows how to stop an online forward recovery (OFR) at the RSR tracking site with a /DBRECOVERY command.

Entry ET (on tracking subsystem):

```
/DISPLAY DATABASE OFR
```

Response ET (to tracking subsystem):

```
DATABASE  TYPE  TOTAL UNUSED  TOTAL UNUSED ACC  CONDITIONS
BE2PCUST  DL/I                             EX   ALLOCS OFR
BE3ORDER  DL/I                             EX   ALLOCS OFR
```

```
  BE3ORDRX  DL/I                              EX   ALLOCS OFR
  BE3PARTS  DL/I                              EX   ALLOCS OFR
  BE3PSID1  DL/I                              EX   ALLOCS OFR
  *91240/132406*   SYS3
```

Entry ET (on tracking subsystem):

```
  /DBRECOVERY DATABASE BE3ORDER BE3PSID1
```

Response ET (to tracking subsystem):

```
  DFS058I DBRECOVERY COMMAND IN PROGRESS
  DFS0488I DBR COMMAND COMPLETED. DBN=BE3ORDER. RC=0
  DFS0488I DBR COMMAND COMPLETED. DBN=BE3PSID1. RC=0
```

Explanation: Databases BE3ORDER and BE3PSID1 are closed, unallocated, and unauthorized with DBRC. Online forward recovery for the databases is also stopped.

Example 4 for /DBRECOVERY Command

This example shows how /DBRECOVERY command might look when entered from the SPOC.

TSO SPOC input:

```
DBR DB BANKATMS BANKTERM BANKLDGR BE3ORDER
```

TSO SPOC output:

```
SYS3     DFS0488I  DBR COMMAND COMPLETED. DBN= BANKATMS RC=04
SYS3     DFS0488I  DBR COMMAND COMPLETED. DBN= BANKTERM RC=04
    SYS3     DFS0488I  DBR COMMAND COMPLETED. DBN= BANKLDGR RC=04
    SYS3     DFS0488I  DBR COMMAND COMPLETED. DBN= BE3ORDER RC= 0
    IMS3     DFS0488I  DBR COMMAND COMPLETED. DBN= BANKATMS RC=04
    IMS3     DFS0488I  DBR COMMAND COMPLETED. DBN= BANKTERM RC=04
    IMS3     DFS0488I  DBR COMMAND COMPLETED. DBN= BANKLDGR RC=04
    IMS3     DFS0488I  DBR COMMAND COMPLETED. DBN= BE3ORDER RC= 0
```

OM API input:

```
CMD (DBR DB BANKATMS BANKTERM BANKLDGR BE3ORDER)
```

OM API output:

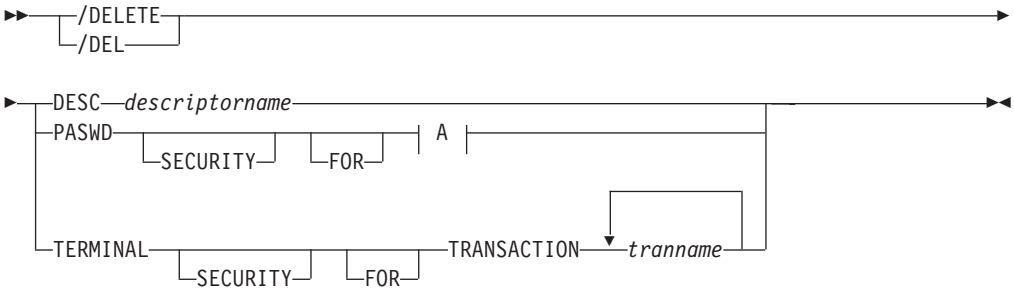
```
<?xml version="1.0"?>
<!DOCTYPE imsout SYSTEM "imsout.dtd">
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2002.197 21:52:02.730359</statime>
<stotime>2002.197 21:52:03.383199</stotime>
<staseq>B7EFBE716A9770A4</staseq>
<stoseq>B7EFBE7209F9FD2F</stoseq>
<rqsttkn1>USRT005 10145202</rqsttkn1>
<rc>0200000C</rc>
<rsn>00003008</rsn>
</ctl>
<cmderr>
<mbr name="SYS3    ">
<typ>IMS     </typ>
<styp>DBDC    </styp>
<rc>00000014</rc>
<rsn>00005050</rsn>
</mbr>
```

```
<mbr name="IMS3    ">
<typ>IMS     </typ>
<styp>DBDC    </styp>
<rc>00000014</rc>
<rsn>00005050</rsn>
</mbr>
</cmderr>
<cmd>
<master>SYS3    </master>
<userid>USRT005 </userid>
<verb>DBR </verb>
<kwd>DB     </kwd>
<input>/DBR DB BANKATMS BANKTERM BANKLDGR BE3ORDER  </input>
</cmd>
<msgdata>
<mbr name="SYS3    ">
 <msg>DFS0488I  DBR COMMAND COMPLETED. DBN= BANKATMS RC=04</msg>
 <msg>DFS0488I  DBR COMMAND COMPLETED. DBN= BANKTERM RC=04</msg>
<msg>DFS0488I  DBR COMMAND COMPLETED. DBN= BANKLDGR RC=04</msg>
<msg>DFS0488I  DBR COMMAND COMPLETED. DBN= BE3ORDER RC= 0</msg>
</mbr>
<mbr name="IMS3    ">
<msg>DFS0488I  DBR COMMAND COMPLETED. DBN= BANKATMS RC=04</msg>
<msg>DFS0488I  DBR COMMAND COMPLETED. DBN= BANKTERM RC=04</msg>
<msg>DFS0488I  DBR COMMAND COMPLETED. DBN= BANKLDGR RC=04</msg>
<msg>DFS0488I  DBR COMMAND COMPLETED. DBN= BE3ORDER RC= 0</msg>
</mbr>
</msgdata>
</imsout>
```

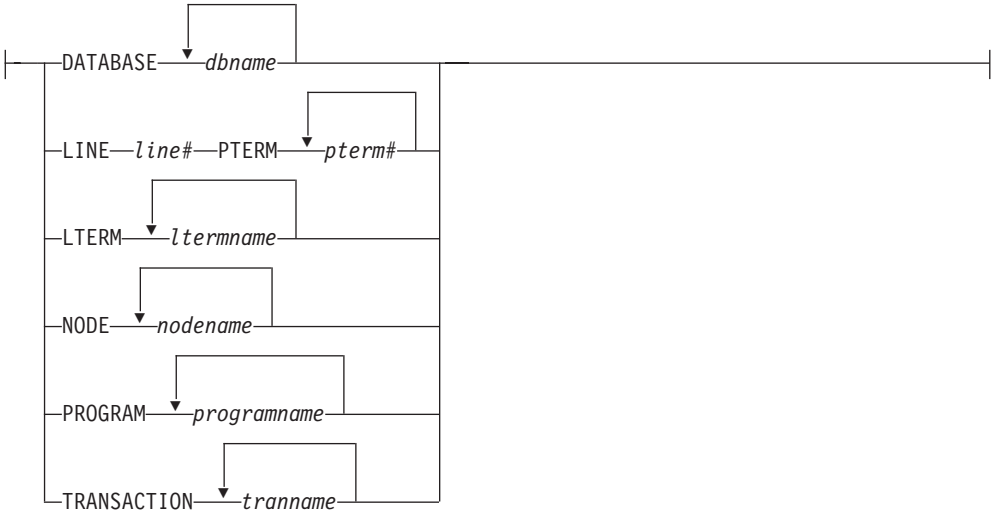
Explanation: The DBR command is routed from OM to the two active IMSs - SYS3 and IMS3. The response from both IMSs is returned to OM. The databases BANKATMS, BANKTERM, BANKLDGR, and BE3ORDER are taken offline at both IMSs.

Chapter 17. /DELETE

Format



A:



Environments and Keywords

Table 36 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 36. Valid Environments for the /DELETE Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/DELETE	X	X	X
DATABASE	X	X	
DESC	X		X
LINE	X		X
LTERM	X		X
NODE	X		X
PASWD	X	X	X
PROGRAM	X	X	X

© Copyright IBM Corp. 1974, 2003

Table 36. Valid Environments for the /DELETE Command and Keywords (continued)

Command / Keywords	DB/DC	DBCTL	DCCTL
PTERM	X		X
TERMINAL	X		X
TRANSACTION	X		X

Usage

The /DELETE command deletes

- LU 6.2 descriptors
- Password security authorizations for a given transaction code, logical terminal, relative physical terminal, database, node, or program
- Terminal and logical link security for a given transaction code

DESCRIPTOR
Specifies the LU62 descriptor to be deleted. Note that this will not delete the descriptor in the PROCLIB member. It is necessary to remove the descriptor from the PROCLIB member; otherwise, the descriptor will be defined at the next IMS restart.

PASSWORD
Specifies the password security authorization to be deleted.

For the /DELETE PASSWORD NODE command, if the specified node is an ISC parallel-session node, password protection is deleted for all the half-sessions of the specified node.

In a dynamic terminal environment, /DELETE PASSWORD is ignored for dynamic ACF-VTAM nodes. This command uses the Security Maintenance utility, which is not supported for dynamic VTAM nodes in a dynamic terminal environment.

Once deleted, password security is not reestablished until either a cold start or a warm start is performed using the PASSWORD keyword.

TERMINAL
Specifies the terminal and logical link security to be deleted. Once deleted, terminal security is not reestablished until either a cold start or a warm start is performed using the TERMINAL keyword.

Examples

Example 1 for /DELETE Command

Entry ET:

```
/DELETE PASSWORD FOR DATABASE TREEFARM
```

Response ET:

```
DFS058I  DELETE COMMAND COMPLETED
```

Explanation: Password security is deleted for the database named TREEFARM.

Example 2 for /DELETE Command

Entry ET:

```
/DELETE TERMINAL SECURITY FOR TRANSACTION PIT, SEED
```

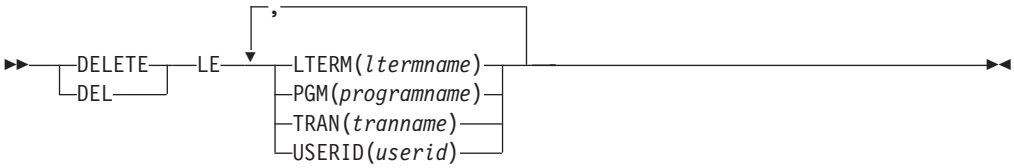
Response ET:

```
DFS058I  DELETE COMMAND COMPLETED
```

Explanation: Terminal and logical link security is deleted for the transaction codes named PIT and SEED.

Chapter 18. DELETE

Format



Environments and Keywords

Table 37 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 37. Valid Environments for the DELETE Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
DELETE LE	X	X	X
LTERM	X	X	X
PGM	X	X	X
TRAN	X	X	X
USERID	X	X	X

Usage

The DELETE command (with the LE keyword) deletes language environment (LE) runtime options that have previously been defined by the UPDATE command (with the LE keyword). The delete can use filters on transaction code, LTERM name, userid, or program name. Any combination of parameters can be used to qualify the application instance to which the runtime parameters are deleted. All matches found are deleted. The entry is logically deleted. The entry is then immediately available for reuse by the next UPDATE LE command. There is a small window of time where it is possible for an entry to be deleted and reused before LE uses it, that could cause LE to use the wrong set of parameters.

Related Reading: For more information about the UPDATE command, see Chapter 65, “UPDATE,” on page 705

This command may be specified only through the Operations Manager API. The command syntax for this command is defined in XML and is available to automation programs that communicate with OM.

OM overrides the routing on the command and routes the command to all IMS systems in the IMSplex. The user-specified route list is ignored.

Rules for matching an entry which results in the delete of an entry:

- If a filter is specified in the command for a particular resource it must match the resource filter defined in the entry. The resource in the DELETE LE command can be specified with wildcards as defined previously.

© Copyright IBM Corp. 1974, 2003

- A resource filter that is not specified on a DELETE LE command will match on any filter for the specific resource defined in the entry. A non-specified filter is treated as a wildcard. For instance if the LTERM filter is not specified on a DELETE LE command it will match on any LTERM resource defined in an entry, as if LTERM(*) was specified on the command.

If an existing set of LE runtime options are found for the specified TRAN, LTERM, USERID, or PGM, the existing parameter string is deleted. The string is deleted only when the specified filters are an exact match for the existing entry. The entry is logically deleted. The entry is available for reuse on the next UPDATE LE command.

The following keywords support a generic or wildcard character. A generic parameter is a 1 - 8 character alphanumeric name that includes one or more special characters and an asterisk or a percent sign. An asterisk can be replaced by a zero or more characters to create a valid resource name. A percent sign can be replaced by exactly one character to create a valid resource name.

LTERM()
Specifies the 1 - 8 character name of the LTERM or LTERMs matching the generic or wildcard parameter.

PGM() Specifies the 1 - 8 character name of the program or programs matching the generic or wildcard parameter.

TRAN()
Specifies the 1 - 8 character name of the transaction or transactions matching the generic or wildcard parameter.

USERID()
Specifies the 1 - 8 character name of the userid or userids matching the generic or wildcard parameter.

Requirement: At least one of the resource filters (LTERM, PGM, TRAN, or USERID) must be specified on the UPDATE LE command.

DELETE LE Output Fields

Table 38 shows the output fields for an DELETE LE command. The columns in the table are as follows:

Short Label
Contains the short label generated in the XML output.

Keyword
Identifies the keyword on the command that caused the field to be generated. N/A appears for output fields that are always returned.

Meaning
Provides a brief description of the output field.

Table 38. Output Fields for the DELETE LE Command

Short Label	Keyword	Meaning
CC	N/A	Completion code for the line of output. Completion code is always returned.
LTRM	LTERM	LTERM name affected by the DELETE .
MBR	N/A	IMSplex member (IMS identifier) that built the output line. Member name is always returned.
PGM	PGM	Program name affected by the DELETE.

Table 38. Output Fields for the DELETE LE Command (continued)

Short Label	Keyword	Meaning
TRAN	TRAN	Transaction name affected by the DELETE.
UID	USERID	Userid affected by the DELETE.

Return, Reason, and Completion Codes for DELETE LE

Table 39 shows the DELETE LE return and reason codes and the meaning of the codes.

Table 39. Return and Reason Codes for the DELETE LE Command

Return Code	Reason Code	Meaning
X'00000000'	X'00000000'	The DELETE LE command completed successfully.
X'00000008'	X'00002008'	No keywords were specified on the command. At least one keyword is required.
X'00000008'	X'0000200C'	No resources found to delete. Either the entry was previously deleted or a keyword filter was typed incorrectly.
X'00000008'	X'00002014'	An invalid character was specified in the filter name.
X'00000010'	X'00004040'	The parameter override header has not been initialized. Retry the command after restart is complete.
X'00000014'	X'00005000'	Unable to get storage from IMODULE GETSTOR.
X'00000014'	X'00005010'	Unable to obtain latch.
X'00000014'	X'00005FFF'	Internal IMS Error - should not occur.

Table 40 contains the completion code that can be returned on a DELETE LE command and the meaning of the code.

Table 40. Completion Code for the DELETE LE Command

Completion Code	Meaning
0	The DELETE LE command completed successfully for the specified resource.

Examples

Example 1 for DELETE LE Command

For this example, Table 41 shows the parameter override table entries prior to any DELETE LE commands being processed.

Table 41. Parameter Override Table Entries for DELETE LE Example

Entry#	TRAN	LTERM	USERID	PROGRAM	LERUNOPTS
1	PART			DFSSAM02	aaaa
2	PART		BETTY		bbbb
3	PART	TERM1			cccc
4				DFSSAM02	dddd
5	PART	TERM1	BARBARA		eeee
6	PART	TERM1	BOB		ffff

Table 41. Parameter Override Table Entries for DELETE LE Example (continued)

Entry#	TRAN	LTERM	USERID	PROGRAM	LERUNOPTS
7		TERM2	BETTY		gggg
8	PART	TERM2			iiii

Table 42 shows the resulting actions when different filters are specified on a DELETE LE command.

Table 42. Example Filters and Resulting Actions for DELETE LE Command

Filters Active on Command	Resulting Action
TRAN(PART)	Deletes entries #1, 2, 3, 5, 6, 8
TRAN(PART) LTERM(TERM1)	Deletes entries #3, 5, 6
LTERM(TERM2) USERID(BETTY)	Deletes entry #7
TRAN(PART) LTERM(TERM1) USERID(BETTY)	Does not delete any entries
TRAN(PART) LTERM(TERM*)	Deletes entries #3, 5, 6, 8
USERID(B*)	Deletes entries #2, 5, 6, 7

Example 2 for DELETE LE Command

The following TSO SPOC and OM API input and output sets up the scenario for the DELETE LE command examples. The UPDATE commands are used to add entries to the table; the QUERY command shows the resulting entries.

TSO SPOC input:

```
UPD LE TRAN(IAPMDI27) USERID(USRT003) SET(LERUNOPTS(cccc))
UPD LE TRAN(IAPMDI27) USERID(USRT001) SET(LERUNOPTS(bbbb))
UPD LE TRAN(IAPMDI26) USERID(USRT001) SET(LERUNOPTS(aaaa))
QRY LE SHOW(ALL)
```

TSO SPOC output (for QUERY command):

```
MbrName    CC Trancode Lterm    Userid   Program  LERunOpts
SYS3        0 IAPMDI26          USRT001           AAAA
SYS3        0 IAPMDI27          USRT001           BBBB
SYS3        0 IAPMDI27          USRT003           CCCC
```

OM API input:

```
CMD(UPD LE TRAN(IAPMDI27) USERID(USRT003) SET(LERUNOPTS(cccc)))
CMD(UPD LE TRAN(IAPMDI27) USERID(USRT001) SET(LERUNOPTS(bbbb)))
CMD(UPD LE TRAN(IAPMDI26) USERID(USRT001) SET(LERUNOPTS(aaaa)))
CMD(QRY LE SHOW(ALL))
```

OM API output (for QUERY command):

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2002.163 18:10:14.974557</statime>
<stotime>2002.163 18:10:14.974985</stotime>
<staseq>B7C4CD747AC5DC85</staseq>
<stoseq>B7C4CD747AE09D25</stoseq>
<rqsttkn1>USRT002 10111014</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
```

```
</ctl>
<cmd>
<master>SYS3    </master>
<userid>USRT002 </userid>
<verb>QRY </verb>
<kwd>LE </kwd>
<input>QRY LE SHOW(ALL) </input>
</cmd>
<cmdrsphdr>
<hdr slbl="MBR"  llbl="MbrName"  scope="LCL" sort="A" key="1" scroll="NO"  len="8"
 dtype="CHAR" align="left" />
<hdr slbl="CC"   llbl="CC"       scope="LCL" sort="N" key="0" scroll="YES" len="4"
 dtype="INT"  align="right" />
<hdr slbl="TRAN" llbl="Trancode" scope="LCL" sort="N" key="0" scroll="YES" len="8"
 dtype="CHAR" align="left" />
<hdr slbl="LTRM" llbl="Lterm"    scope="LCL" sort="N" key="0" scroll="YES" len="8"
 dtype="CHAR" align="left" />
<hdr slbl="UID"  llbl="Userid"   scope="LCL" sort="N" key="0" scroll="YES"  len="8"
 dtype="CHAR" align="left" />
<hdr slbl="PGM"  llbl="Program"  scope="LCL" sort="N" key="0" scroll="YES" len="8"
 dtype="CHAR" align="left" />
<hdr slbl="PRM"  llbl="LERunOpts" scope="LCL" sort="N" key="0" scroll="YES" len="*"
 dtype="CHAR" align="left" />
</cmdrsphdr>
<cmdrspdata>
<rsp>MBR(SYS3    ) CC(   0) TRAN(IAPMDI26) LTRM(      ) UID(USRT001 )  PGM(      )
PRM(AAAA    ) </rsp>
<rsp>MBR(SYS3    ) CC(   0) TRAN(IAPMDI27) LTRM(      ) UID(USRT001 )  PGM(      )
 PRM(BBBB    ) </rsp>
<rsp>MBR(SYS3    ) CC(   0) TRAN(IAPMDI27) LTRM(      ) UID(USRT003 )  PGM(      )
 PRM(CCCC    ) </rsp>
</cmdrspdata>
</imsout>
```

DELETE LE command example number 1.

TSO SPOC input:

```
DEL LE TRAN(IAPMDI27) USERID(USRT001)
```

TSO SPOC output:

```
MbrName    CC Trancode Lterm    Userid   Program
SYS3        0 IAPMDI27          USRT001
```

OM API input:

```
CMD(DEL LE TRAN(IAPMDI27) USERID(USRT001))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2002.163 18:12:21.932813</statime>
<stotime>2002.163 18:12:21.933334</stotime>
<staseq>B7C4CDED8E70D786</staseq>
<stoseq>B7C4CDED8E916246</stoseq>
<rqsttkn1>USRT002 10111221</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<master>SYS3    </master>
<userid>USRT002 </userid>
<verb>DEL </verb>
```

```
<kwd>LE              </kwd>
<input>DEL LE TRAN(IAPMDI27) USERID(USRT001) </input>
</cmd>
<cmdrsphdr>
<hdr slbl="MBR"  llbl="MbrName"  scope="LCL" sort="A" key="1" scroll="NO"  len="8"
 dtype="CHAR" align="left" />
<hdr slbl="CC"   llbl="CC"       scope="LCL" sort="N" key="0" scroll="YES" len="4"
 dtype="INT" align="right" />
<hdr slbl="TRAN" llbl="Trancode" scope="LCL" sort="N" key="0" scroll="YES" len="8"
 dtype="CHAR" align="left" />
<hdr slbl="LTRM" llbl="Lterm"    scope="LCL" sort="N" key="0" scroll="YES" len="8"
 dtype="CHAR " align="left" />
<hdr slbl="UID"  llbl="Userid"   scope="LCL" sort="N" key="0" scroll="YES" len="8"
 dtype="CHAR " align="left" />
<hdr slbl="PGM"  llbl="Program"  scope="LCL" sort="N" key="0" scroll="YES" len="8"
 dtype="CHAR" align="left" />
</cmdrsphdr>
<cmdrspdata>
<rsp>MBR(SYS3    ) CC(   0) TRAN(IAPMDI27) LTRM(      ) UID(USRT001 )  PGM(      )
</rsp>
</cmdrspdata>
</imsout>
```

Explanation: This command specifies two filters- transaction and userid. A table entry is deleted if both of these filters are defined and match what was specified in the command. This command deletes one entry from the table. That entry is shown in the command output.

DELETE LE command example number 2.

TSO SPOC input:

```
DEL LE USERID(USRT00%)
```

TSO SPOC output:

```
MbrName    CC Trancode Lterm    Userid   Program
SYS3        0 IAPMDI26          USRT001
SYS3        0 IAPMDI27          USRT003
```

OM API input:

```
CMD(DEL LE USERID(USRT00%))
```

OM API output:

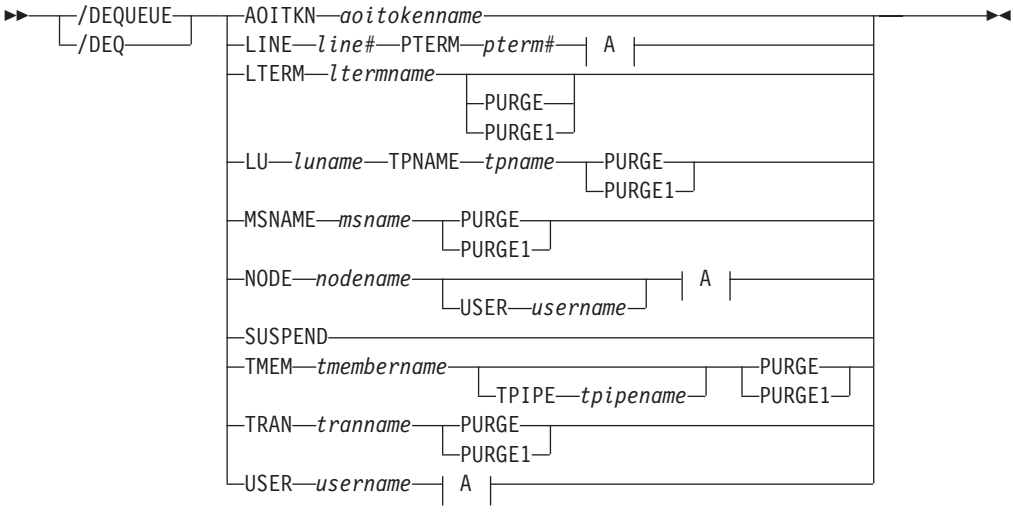
```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2002.163 18:19:45.616468</statime>
<stotime>2002.163 18:19:45.616972</stotime>
<staseq>B7C4CF94AFA54304</staseq>
<stoseq>B7C4CF94AFC4C704</stoseq>
<rqsttkn1>USRT002 10111945</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<master>SYS3    </master>
<userid>USRT002 </userid>
<verb>DEL </verb>
<kwd>LE              </kwd>
<input>DEL LE USERID(USRT00%)    </input>
</cmd>
<cmdrsphdr>
```

```
<hdr slbl="MBR" llbl="MbrName" scope="LCL" sort="A" key="1" scroll="NO"  len="8"
 dtype="CHAR" align="left" />
<hdr slbl="CC" llbl="CC" scope="LCL" sort="N" key="0" scroll="YES" len="4"
 dtype="INT" align="right" />
<hdr slbl="TRAN" llbl="Trancode" scope="LCL" sort="N" key="0" scroll="YES" len="8"
 dtype="CHAR" align="left" />
<hdr slbl="LTRM" llbl="Lterm" scope="LCL" sort="N" key="0" scroll="YES"  len="8"
 dtype="CHAR " align="left" />
<hdr slbl="UID" llbl="Userid" scope="LCL" sort="N" key="0" scroll="YES"  len="8"
 dtype="CHAR " align="left" />
<hdr slbl="PGM" llbl="Program" scope="LCL" sort="N" key="0" scroll="YES" len="8"
 dtype="CHAR" align="left" />
</cmdrsphdr>
<cmdrspdata>
<rsp>MBR(SYS3    ) CC(   0) TRAN(IAPMDI26) LTRM(      ) UID(USRT001 )  PGM(      )
</rsp>
<rsp>MBR(SYS3    ) CC(   0) TRAN(IAPMDI27) LTRM(      ) UID(USRT003 )  PGM(      )
</rsp>
</cmdrspdata>
</imsout>
```

Explanation: This command specifies only one filter, userid. A wildcard is used, so any table entry where the USERID filter matches the string will be deleted. The command output shows that both of the remaining table entries were deleted.

Chapter 19. /DEQUEUE

Format



A:

Environments and Keywords

Table 43 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 43. Valid Environments for the /DEQUEUE Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/DEQUEUE	X	X	X
AOITKN	X	X	X
LINE	X		X
LTERM	X		X
LU	X		X
MSNAME	X		X
NODE	X		X
PTERM	X		X
PURGE	X		X
PURGE1	X		X
SUSPEND	X		X
TMEM	X		X

© Copyright IBM Corp. 1974, 2003

Table 43. Valid Environments for the /DEQUEUE Command and Keywords (continued)

Command / Keywords	DB/DC	DBCTL	DCCTL
TPIPE	X		X
TPNAME	X		X
TRAN	X		X
USER	X		X

Usage

The /DEQUEUE command with any of the following keywords is used to cancel a message that is being processed: LINE, LTERM, LUNAME, MSNAME, NODE, TRANSACTION, or USER. Also, a /DEQUEUE command dequeues messages from either the local queues, or in a shared-queues environment, the shared queues. An output message displays the total count of all messages dequeued.

/DEQUEUE AOITOKEN is used to dequeue and discard messages associated with the specified AOI token name.

A /DEQUEUE command with any of the following keywords activates the Message Control Error exit routine, DFSCMUX0, before processing each message: LTERM, LUNAME TPNAME, NODE, or MSNAME. The user exit routine can request IMS to:

- Proceed with the command and purge the message. This is the default action for the Message Control Error exit routine. For more information, see *IMS Version 9: Customization Guide*.
- Suppress the command.
- Suppress the command and send an informational message to the entering terminal.
- Route the message to a different destination.

/DEQUEUE SUSPEND schedules transactions that have been suspended by IMS.

/DEQUEUE TMEMBER TPIPE dequeues messages associated with an IMS Open Transaction Manager Access (OTMA) transaction pipe.

The /DEQUEUE NODE command dequeues messages from the local or shared queues for statically defined terminals. If PURGE or PURGE1 is specified, the node must be stopped and idle. Without PURGE or PURGE1, the message actively being sent is dequeued.

The /DEQUEUE LTERM command dequeues messages from the local or shared queues. The user must be stopped (by a /STOP USER command), and not in conversation, in order to enter /DEQUEUE USER or /DEQUEUE LTERM with PURGE or PURGE1. In a shared queues and ETO environment where the user doesn't exist, /DEQUEUE LTERM with PURGE or PURGE1 creates a user structure in order to dequeue messages from the shared queues.

AOITOKEN
Dequeues and discards all messages associated with the specified AOI token name.

LINE PTERM
Cancels the output message currently being sent to the specified physical terminal on the specified communication line.

LTERM
Cancels the output message currently being sent to the specified logical terminal.

`/DEQUEUE NODE nodename LTERM` is not valid for nodes with users. The user that contains the LTERM must be allocated and the session must be active.

If the PURGE or PURGE1 keyword is specified and the LTERM is associated with a user with DEADQ status, the DEADQ status will be reset.

If the DEADQ status is on, the following command turns it off for the user associated with LTERM ltermname:

```
/DEQUEUE LTERM ltermname (PURGE/PURGE1)
```

The DEADQ status is removed by signing the user on or by issuing `/DEQUEUE` to dequeue one or more messages.

LUNAME TPNAME
Specifies the LU name and TP name of the LU 6.2 destination from which messages are discarded. A normal (non-qualified) or fully network qualified LUNAME is required for the LUNAME keyword.

Either the PURGE or PURGE1 option must be specified, and the LU 6.2 destination must be stopped and not active; if both actions are not taken, error messages are issued.

MSNAME
Cancels the output message currently being sent to the specified MSC link. Either PURGE or PURGE1 is required for this link. The MSC link must be stopped by the `/PSTOP LINK command` before issuing the `/DEQUEUE MSNAME` command.

NODE
Cancels the output message currently being sent to the specified VTAM node.

The following command is not valid for nodes with users:

```
/DEQUEUE NODE nodename
/DEQUEUE NODE nodename LTERM ltermname
```

The following command cancels the output message currently being sent to the specified ISC session, and the user must be allocated to the node and the session must be active:

```
/DEQUEUE NODE nodename USER username
```

If PURGE or PURGE1 is specified, the node must be stopped and idle. Without PURGE and PURGE1, if an ETO node or an ISC static parallel session, the USER keyword is required.

If the PURGE keyword is specified, and the NODE is associated with a user with DEADQ status, the DEADQ status will be reset.

PURGE
Cancels all output messages enqueued to the specified LINE, LTERM, LUNAME, MSNAME, NODE, TMEMBER, TPIPE, or USER. `PURGE` allows cancellation of all output messages enqueued to an LTERM, or to a PTERM without regard to the LTERM. A `/DEQUEUE` command without PURGE cancels an in-progress output message on the specified terminal and, optionally, LTERM. If the NODE and LTERM keywords have both been supplied, the LTERM must be a valid LTERM of the user currently associated with the node.

Messages cannot be canceled with PURGE if they are destined for the MTO or for terminals or users in response mode, Fast Path, or conversation mode. /DEQUEUE PURGE does not cancel messages destined for the master terminal operator because the master terminal cannot be stopped, disconnected, or idled. /EXIT should be used for a terminal in conversation mode.

If the terminal is a dynamic 3600/FINANCE, SLU P, or ISC terminal, the NODE keyword is only valid if the session is stopped and idle, and the LTERMs are still allocated to the terminal pending message resynchronization. For example,

```
/DEQUEUE NODE nodename USER username (LTERM ltermname) PURGE
```

is valid for the allocated user only if the specified ISC session is stopped and idled.

Messages queued to LTERMs associated with users that are not signed on can be purged by specifying the LTERM keyword without the NODE keyword. NODE with the USER keyword applies to ISC sessions and dynamic terminals and is only valid if the user is still allocated to the node. However, the line or node must be stopped and idled.

If USER is specified and the user had DEADQ status, the DEADQ status is removed. If LTERM or NODE is specified and the LTERM or node is associated with a user with DEADQ status, the DEADQ status is removed.

PURGE1

Cancels the first message queued to the specified LTERM. The PURGE1 and PURGE keywords are mutually exclusive.

/DEQUEUE LTERM PURGE1 removes the first message from the queue. The order of search for messages to be removed is:

1. Fast Path
2. Response mode
3. Exclusive mode messages
4. System messages
5. All other messages

Using a synonym for PURGE1 avoids the potential danger of omitting the trailing digit on PURGE1 and canceling all the messages enqueued. Synonyms for PURGE1 are SINGLE, SNGL, and FIRST.

When PURGE1 is specified, the terminal must be stopped and idle and not in conversation mode. If the message is in the process of being actively sent, /DEQUEUE without PURGE1 cancels the message.

SUSPEND

Reschedules transactions that have been suspended by IMS. SUSPEND applies to IMS systems sharing data at the block level and to connected external subsystems (not CCTL subsystems). /DEQUEUE SUSPEND causes all message-driven transactions that have been suspended to be rescheduled. When a failed IMS system is restarted, /DEQUEUE SUSPEND informs the currently running IMS system to reschedule any message-driven transactions on its SUSPEND queue. However, when the failed subsystem has global locks retained and cannot communicate the release of those locks to the other subsystems, /DEQUEUE SUSPEND is used to reschedule these message-driven transactions in the other subsystems. This can occur when:

- The still-running IMS has transactions on its SUSPEND queue and the failed IMS has been restarted using the /ERESTART COLDSYS command.
- The still-running IMS terminates before the failed IMS is restarted. Thus, when the failed system is restarted, the other system, which has now terminated, is unable to receive the message to reschedule the transactions on its SUSPEND queue. The operator must issue /DEQUEUE SUSPEND to reschedule them.

Another use for /DEQUEUE SUSPEND is to release those transactions queued because the installation specified the requeue region error option in the external subsystem PROCLIB member for a particular subsystem. Transactions will be queued if the subsystem connection terminates in an abnormal fashion (for example, dependent region abend, subsystem abnormal termination).

In a shared-queues environment, a /DEQUEUE SUSPEND command moves all transactions on the Transaction Suspend queue to the Transaction Ready queue and makes them eligible for rescheduling. The “suspend” status for the transactions is also reset locally (on the IMS subsystem that entered the command), but is not reset across the sysplex. To reset the “suspend” status across a sysplex, use the /START TRANSACTION command on each IMS subsystem.

You can also use the /START TRANSACTION command to reschedule a specific transaction that has been suspended.

/DISPLAY TRANSACTION and /DISPLAY STATUS can be used to determine whether a transaction has messages on its suspend queue.

TMEMBER

Dequeues messages from the OTMA transaction pipe structure specified as *tpipename*. Using the /DEQ command can make OTMA message resynchronization impossible. And if resynchronization is in progress for the specified transaction pipe, IMS rejects the command and issues message DFS2392I.

TPIPE

Is used with the /DEQUEUE command to dequeue messages associated with an OTMA transaction pipe.

TRANSACTION

Dequeues one or more messages from the transaction specified. The transaction name is a 1- to 8-byte name. The transaction must be stopped for messages to be dequeued.

Restriction: This keyword is valid only in a shared-queues environment.

Recommendation: Stop the transaction on all IMS subsystems in the shared queues group before issuing the /DEQUEUE TRANSACTION command. If you do not, the IMS subsystem that processes the /DEQUEUE could dequeue a message it does not own and that another IMS subsystem might need.

A /DEQUEUE TRANSACTION command does not dequeue messages enqueued for suspended transactions.

USER (LTERM)

Dequeues all messages associated with all LTERMs (or a specific) LTERM assigned to that USER. The user must be stopped and not in conversation in order to enter /DEQUEUE USER with PURGE and PURGE1. /DEQUEUE USER with PURGE1 or /DEQUEUE USER LTERM without PURGE or PURGE1 requires the

LTERM keyword. If a user is signed on, /DEQUEUE USER LTERM can be used to cancel an output message that is currently being processed. When the purge options are used, messages destined for the MTO and users in response mode, Fast Path, and conversation cannot be canceled. If the USER and LTERM keywords have both been supplied, the LTERM must be a valid LTERM of the specified USER.

If the user has DEADQ status, the DEADQ status is removed.

Examples

Example 1 for /DEQUEUE Command

Entry ET:

```
/DEQUEUE AOITOKEN ABCDE
```

Response ET:

```
DFS058I DEQUEUE COMMAND COMPLETED
```

Explanation: The messages on AOITOKEN ABCDE are dequeued and discarded.

Example 2 for /DEQUEUE Command

Entry ET:

```
/DEQUEUE LINE 5 PTERM 2
```

Response ET:

```
DFS058I  DEQUEUE COMMAND COMPLETED
```

Explanation: The message is terminated on successful completion of the current transmission.

Example 3 for /DEQUEUE Command

Entry ET:

```
/DEQUEUE LTERM ABCDE PURGE
```

Response ET:

```
DFS976I 2 MESSAGE(S) PURGED
```

Explanation: Two messages are dequeued for LTERM ABCDE (the line is stopped and idle as required).

Example 4 for /DEQUEUE Command

Entry ET:

```
/DEQUEUE LTERM ABCDE PURGE1
```

Response ET:

```
DFS058I  DEQUEUE COMMAND COMPLETED
```

Explanation: The message is dequeued for LTERM ABCDE.

Example 5 for /DEQUEUE Command

Entry ET:

```
/DEQ LUNAME 'NETWORK1.LUNAME1' TPNAME TPNAME1 PURGE
```

Response ET:

```
DFS1952 10:51:35 DESTINATION NOT FOUND
```

Explanation: An invalid APPC destination was entered.

Example 6 for /DEQUEUE Command

Entry ET:

```
/DEQ LUNAME 'CICSNET.SERVER1' TPNAME DFSASYNC PURGE1
```

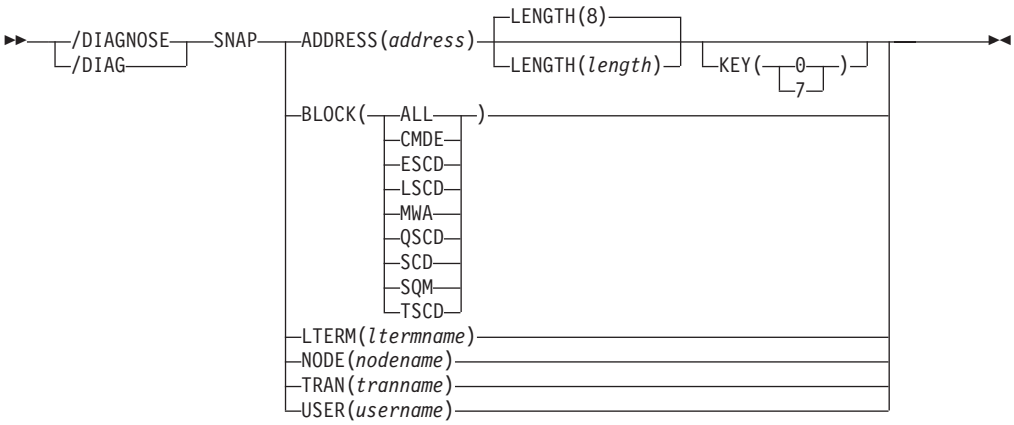
Response ET:

```
DFS976I 14:10:25      1 MESSAGE(S) PURGED
```

Explanation: The message was dequeued from the APPC destination LUNAME CICSNET.SERVER, TPNAME DFSASYNC.

Chapter 20. /DIAGNOSE

Format



Environments

Table 44 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command can be issued.

Table 44. Valid Environments for the /DIAGNOSE Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/DIAGNOSE	X	X	X
ADDRESS	X	X	X
BLOCK	X	X	X
LTERM	X		X
NODE	X		X
SNAP	X	X	X
TRAN	X		X
USER	X		X

Usage

The /DIAGNOSE command allows users to retrieve diagnostic information for system resources such as IMS control blocks, user-defined nodes, or user-defined transactions at any time without taking a console dump.

SNAP Captures storage information and writes the information to the OLDS as type X'6701' records. Storage information can be captured for IMS control blocks, user-defined nodes, and user-defined transactions.

ADDRESS
Captures information about a specific area stored in memory. The *address* identifies the area and must specify a hexadecimal value between 0 and 7FFFFFFF.

LENGTH
Specifies the length of the address. The LENGTH parameter is

© Copyright IBM Corp. 1974, 2003

optional and has a default value of 8. If the LENGTH parameter is specified, the *length* must be a numeric value between 1 and 65536.

KEY
Alters the PSW key prior to accessing the specified storage. The KEY parameter is optional. Valid values for the KEY parameter must be a numeric value between 0 and 7.

BLOCK
Captures information for a specific IMS control block. A control block name or ALL is required. Valid control block names include:

ALL Captures information for all valid control blocks currently available.

CMDE Captures information for the Commands SCD Extension control block.

ESCD Captures information for the Extended System Contents Directory control block.

LSCD Captures information for the LU 6.2 Extension to SCD control block.

MWA Captures information for the Modify Work Area control block.

QSCD Captures information for the Queue Manager Extension to SCD control block.

SCD Captures information for the System Contents Directory control block.

SQM Captures information for the Shared Queue Master control block.

TSCD Captures information for the OTMA Extension to SCD control block.

LTERM
Captures information about the logical terminal specified in the parameter, *ltermname*. If an LTERM is locally defined, the local control block information is captured. In a shared queues environment, if an LTERM is not defined on the local system issuing the command, global information for that LTERM is gathered from an available Resource Structure. If a control block is available, information associated with the specified LTERM is captured. Following is a list of control blocks:

CLB Communication Line Block

CTB Communication Terminal Block

CTT Communication Translate Table

CRB Communications Restart Block

SPQB Subpool Queue Block

CNT Communication Name Table (maximum of five)

CCB Conversational Control Block

CIB Communication Interface Block

	INBUF	Input Line Buffer
	OUTBF	Output Line Buffer

NODE Captures information about the node specified in the parameter, *nodename*. If a NODE is locally defined, the local NODE information is captured. In a shared queues environment, if a NODE is not defined on the local system issuing the command, global information for that NODE is gathered from an available Resource Structure. If a control block is available, information associated with the specified NODE is captured. Following is a list of control blocks:

	CLB	Communication Line Block
	CTB	Communication Terminal Block
	CTT	Communication Translate Table
	CRB	Communications Restart Block
	SPQB	Subpool Queue Block
	CNT	Communication Name Table (maximum of five)
	CCB	Conversational Control Block
	CIB	Communication Interface Block
	INBUF	Input Line Buffer
	OUTBF	Output Line Buffer

TRAN Captures information about the transaction specified in the parameter, *tranname*. This keyword captures the SMB data for a specified transaction.

USER Captures information about the user specified in the parameter, *username*. If a USER is locally defined, the local USER information is captured. In a shared queues environment, if a USER is not defined on the local system issuing the command, global information for that USER is gathered from an available Resource Structure. If a control block is available, information associated with the specified USER is captured. Following is a list of control blocks:

	CLB	Communication Line Block
	CTB	Communication Terminal Block
	CTT	Communication Translate Table
	CRB	Communications Restart Block
	SPQB	Subpool Queue Block
	CNT	Communication Name Table (maximum of five)
	CCB	Conversational Control Block
	CIB	Communication Interface Block
	INBUF	Input Line Buffer
	OUTBF	Output Line Buffer

Example for /DIAGNOSE Command

Entry ET:

```
/DIAGNOSE SNAP TRAN(tranname)
```

Response ET:

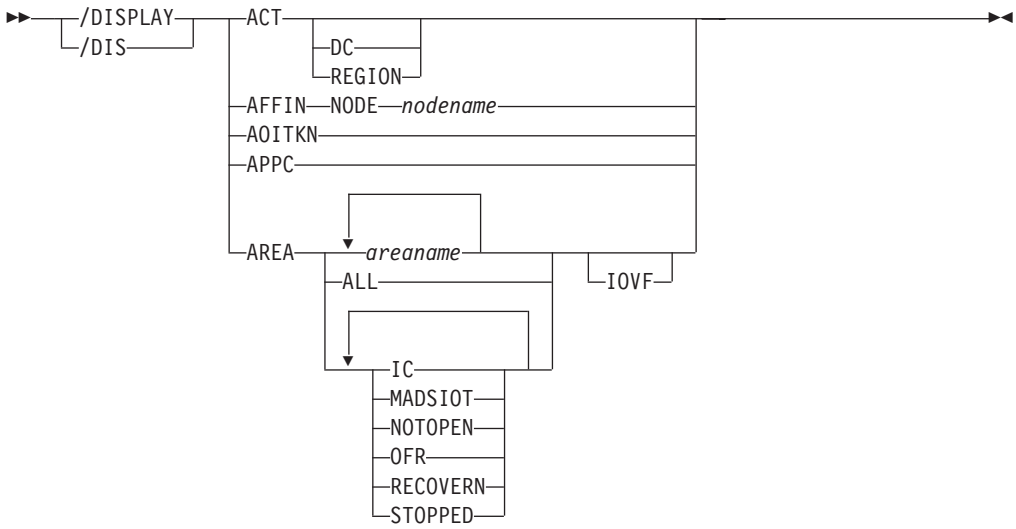
```
DFS058I DIAGNOSE COMMAND COMPLETED
```

Explanation: Diagnostic information about transaction tranname is captured on the IMS log.

Chapter 21. /DISPLAY

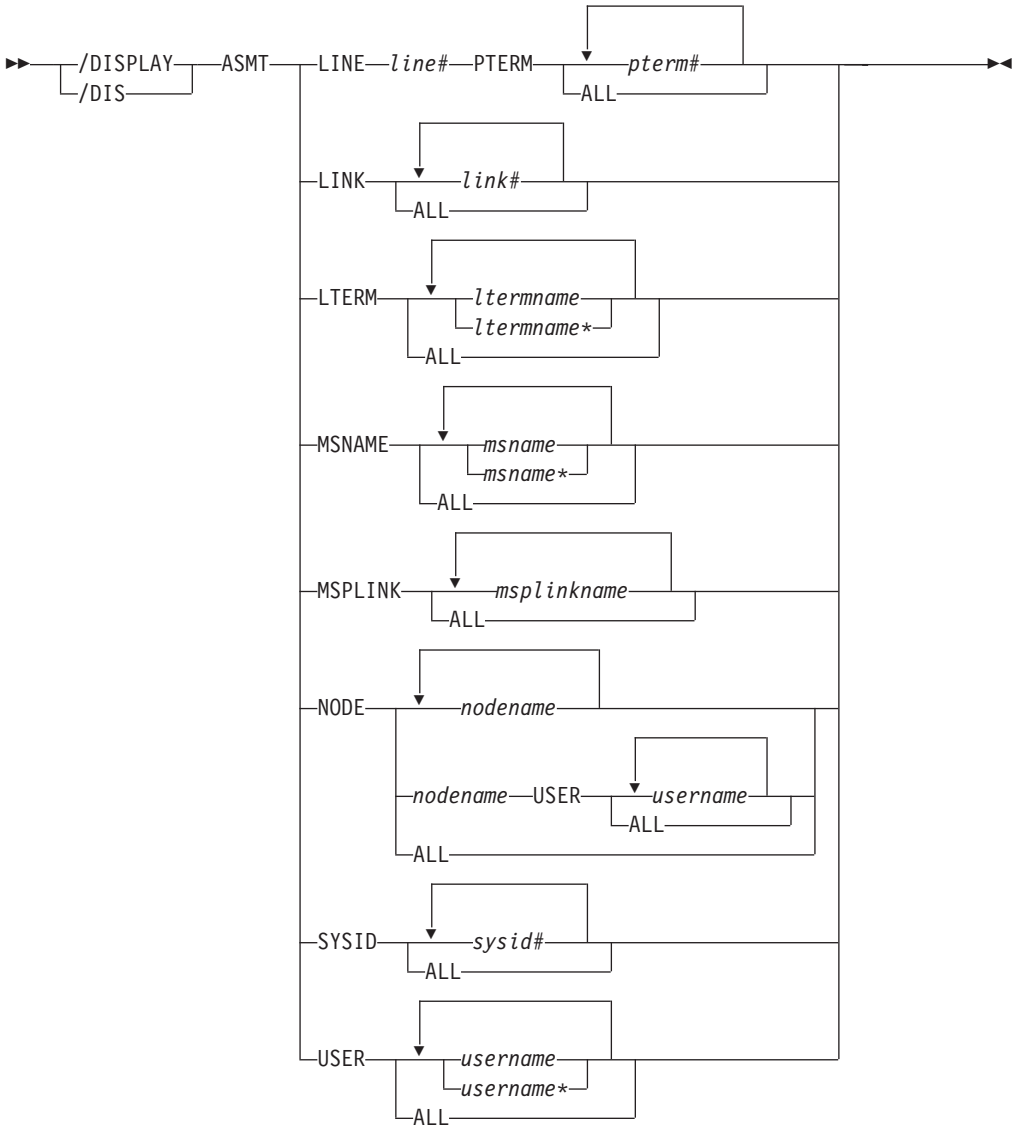
Format

/DISPLAY ACTIVE Command

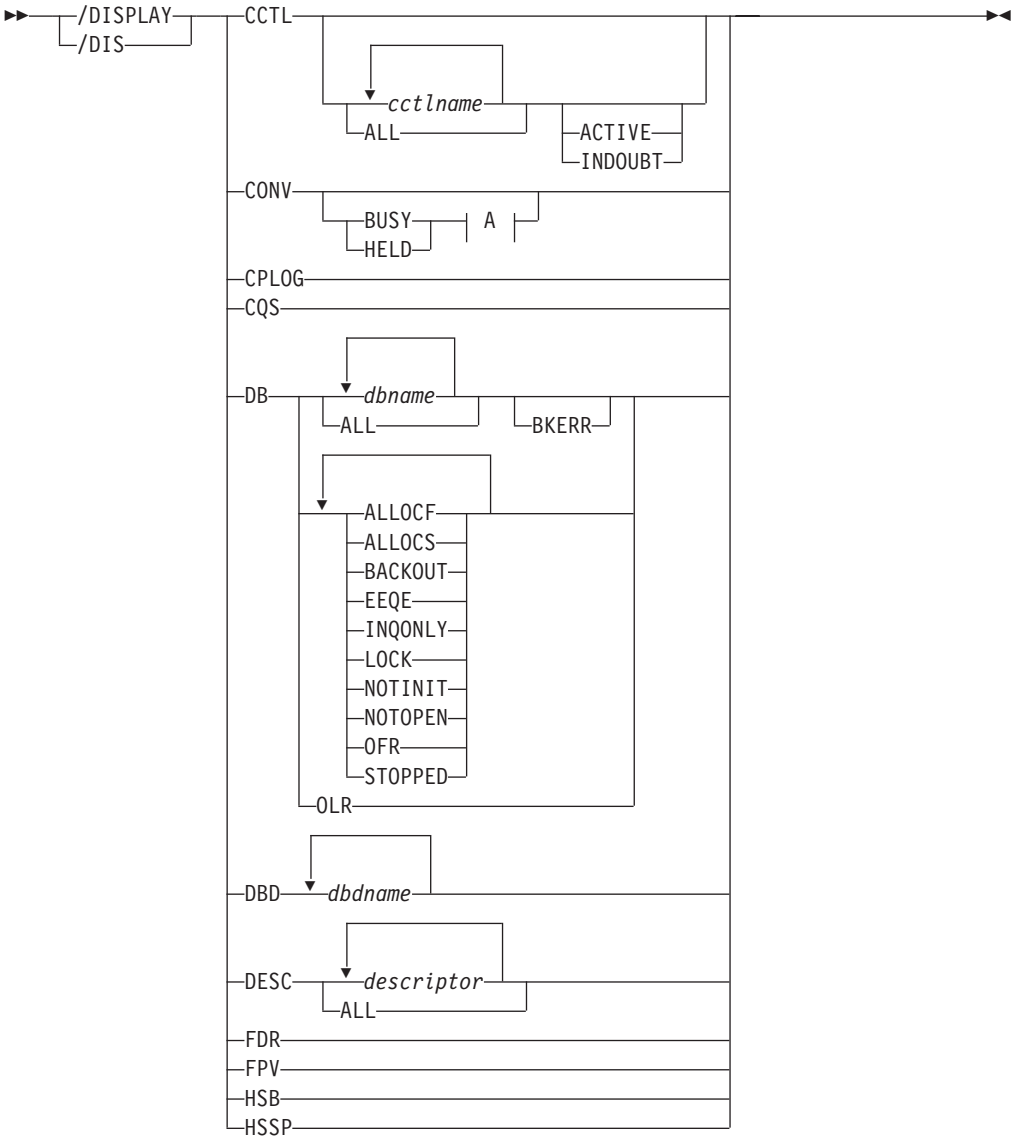


© Copyright IBM Corp. 1974, 2003

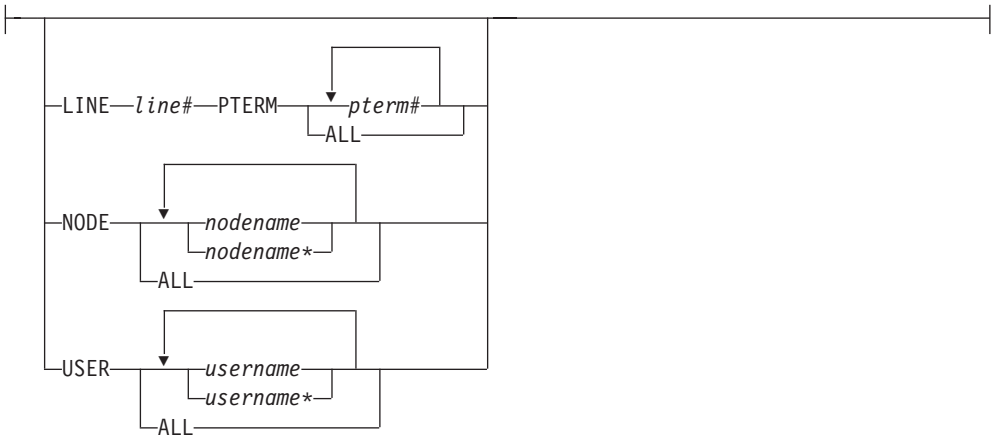
/DISPLAY ASSIGNMENT Command



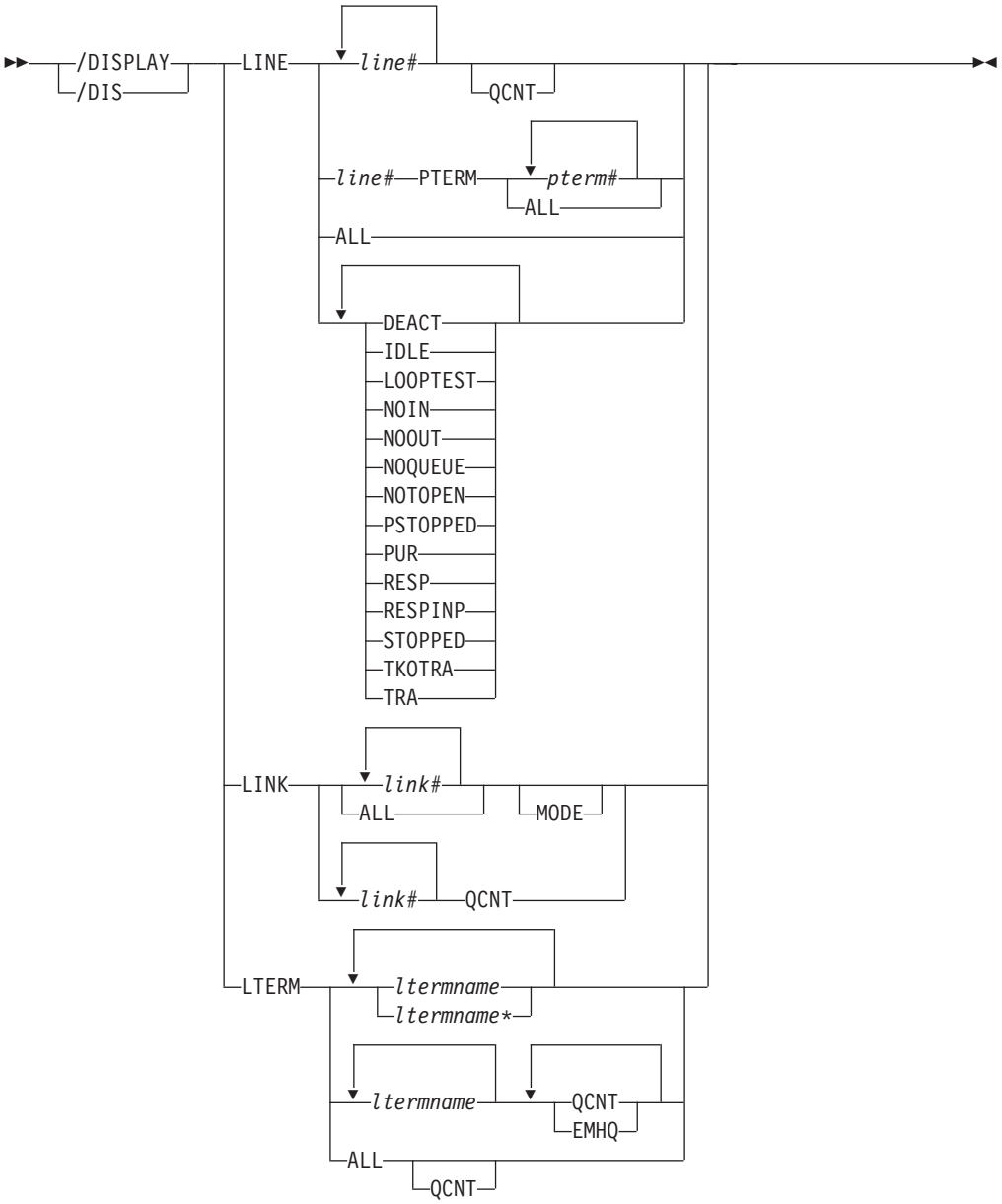
/DISPLAY Command: CCTL Through HSSP



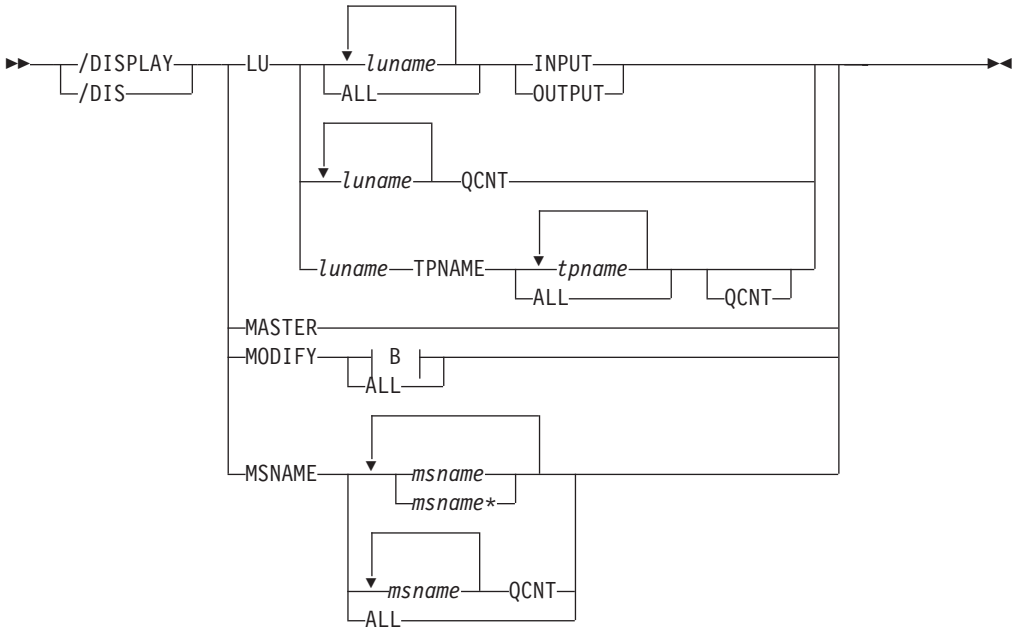
A:


LINE
line#
PTERM
pterm#
ALL
NODE
nodename
nodename*
ALL
USER
username
username*
ALL

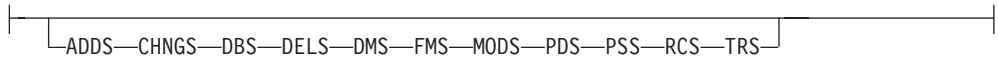
/DISPLAY Command: LINE through LTERM



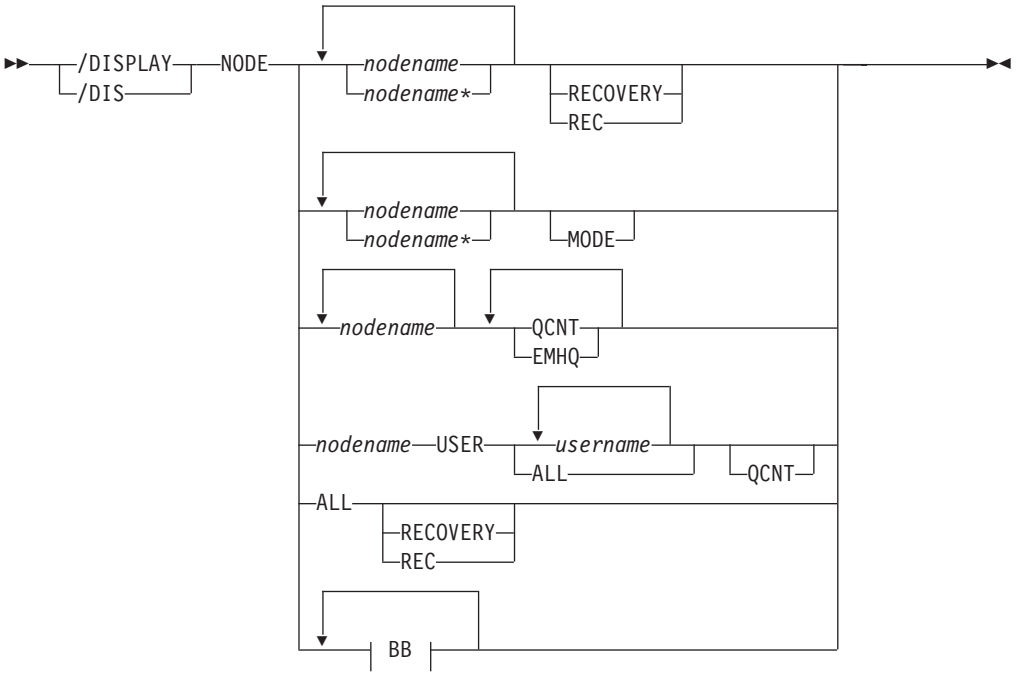
/DISPLAY Command: LU through MSNAME



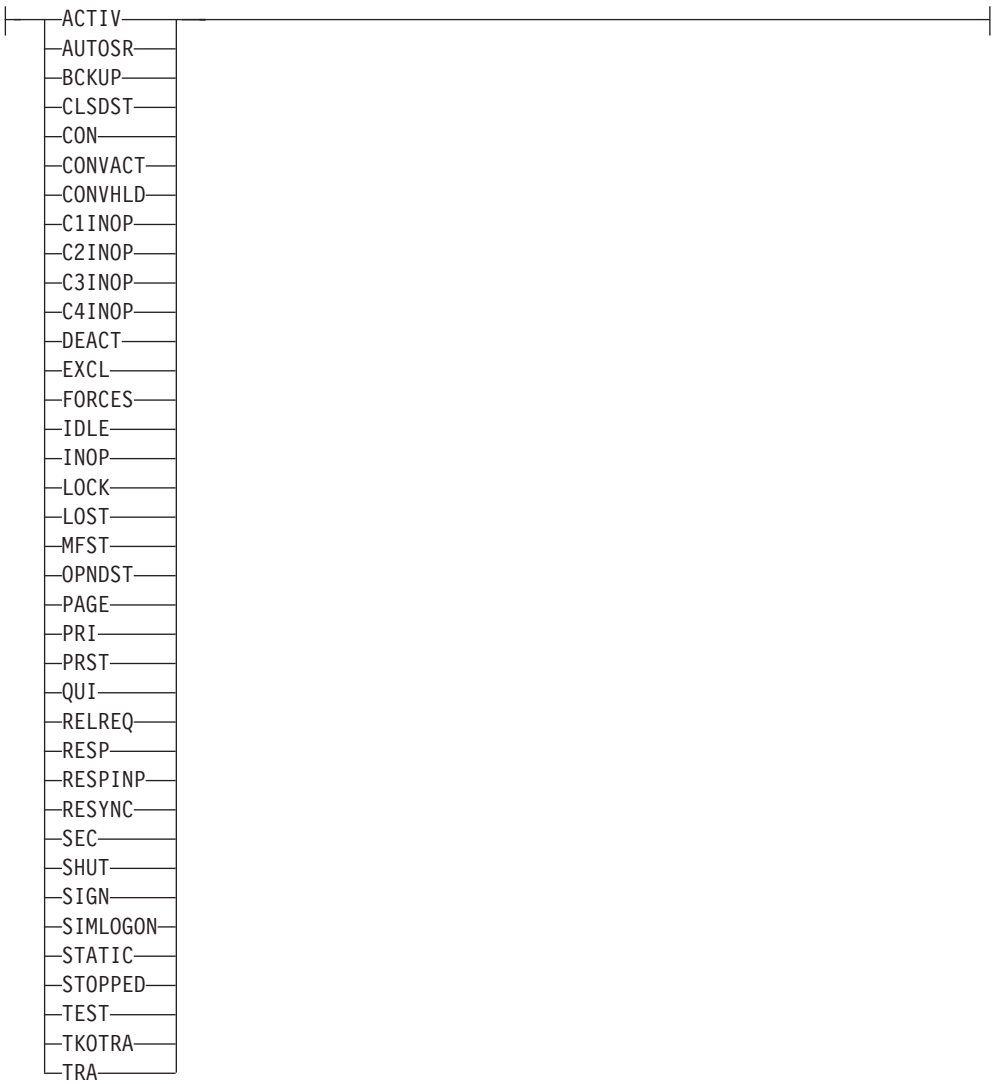
B:



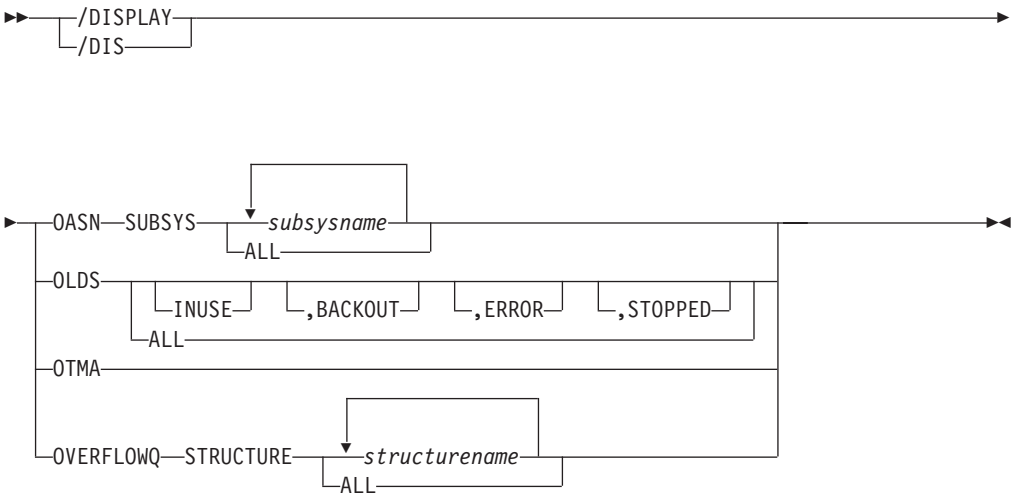
/DISPLAY NODE Command



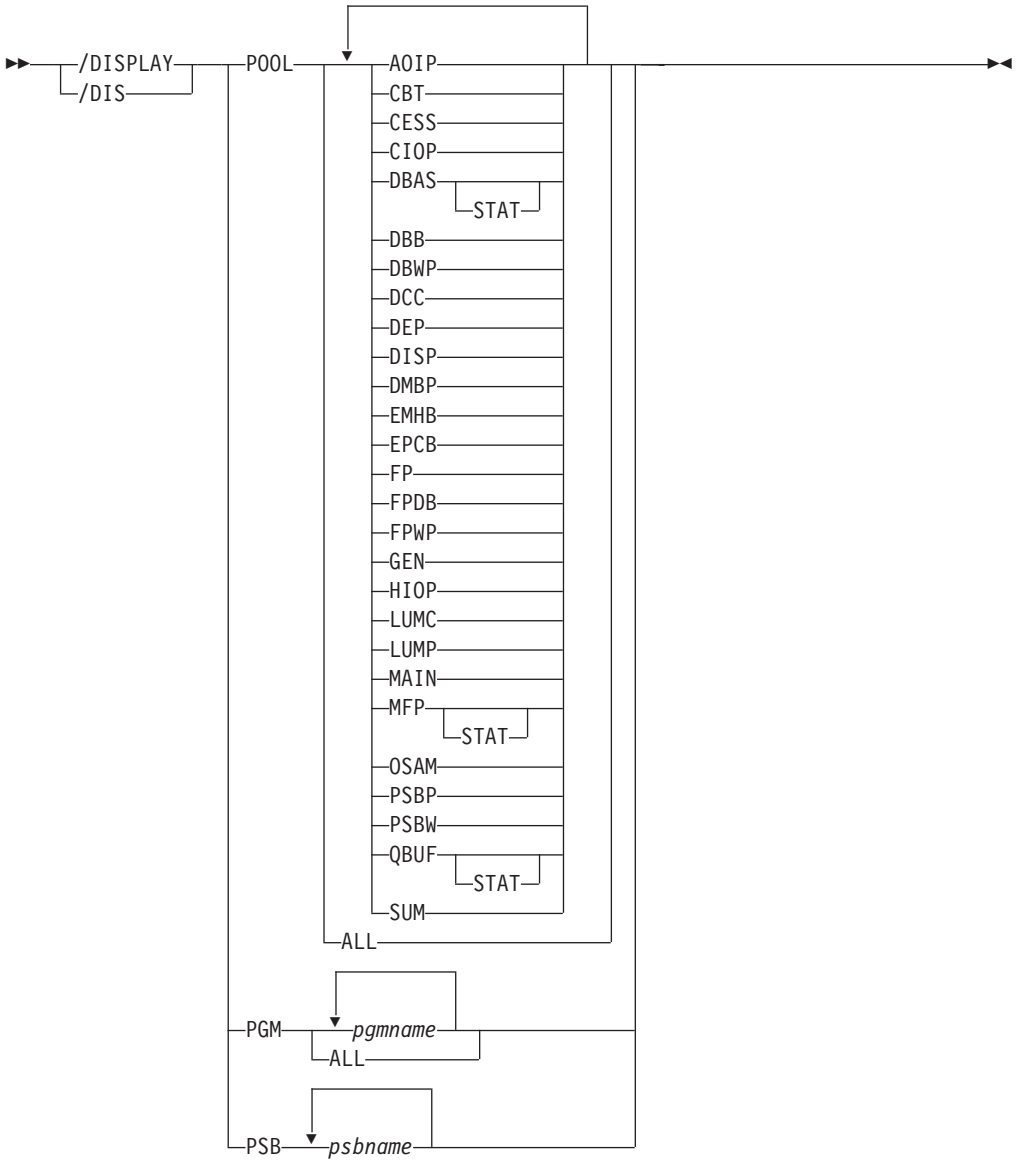
BB:


ACTIV
AUTOSR
BCKUP
CLSDST
CON
CONVACT
CONVHLD
C1INOP
C2INOP
C3INOP
C4INOP
DEACT
EXCL
FORCES
IDLE
INOP
LOCK
LOST
MFST
OPNDST
PAGE
PRI
PRST
QUI
RELREQ
RESP
RESPINP
RESYNC
SEC
SHUT
SIGN
SIMLOGON
STATIC
STOPPED
TEST
TKOTRA
TRA

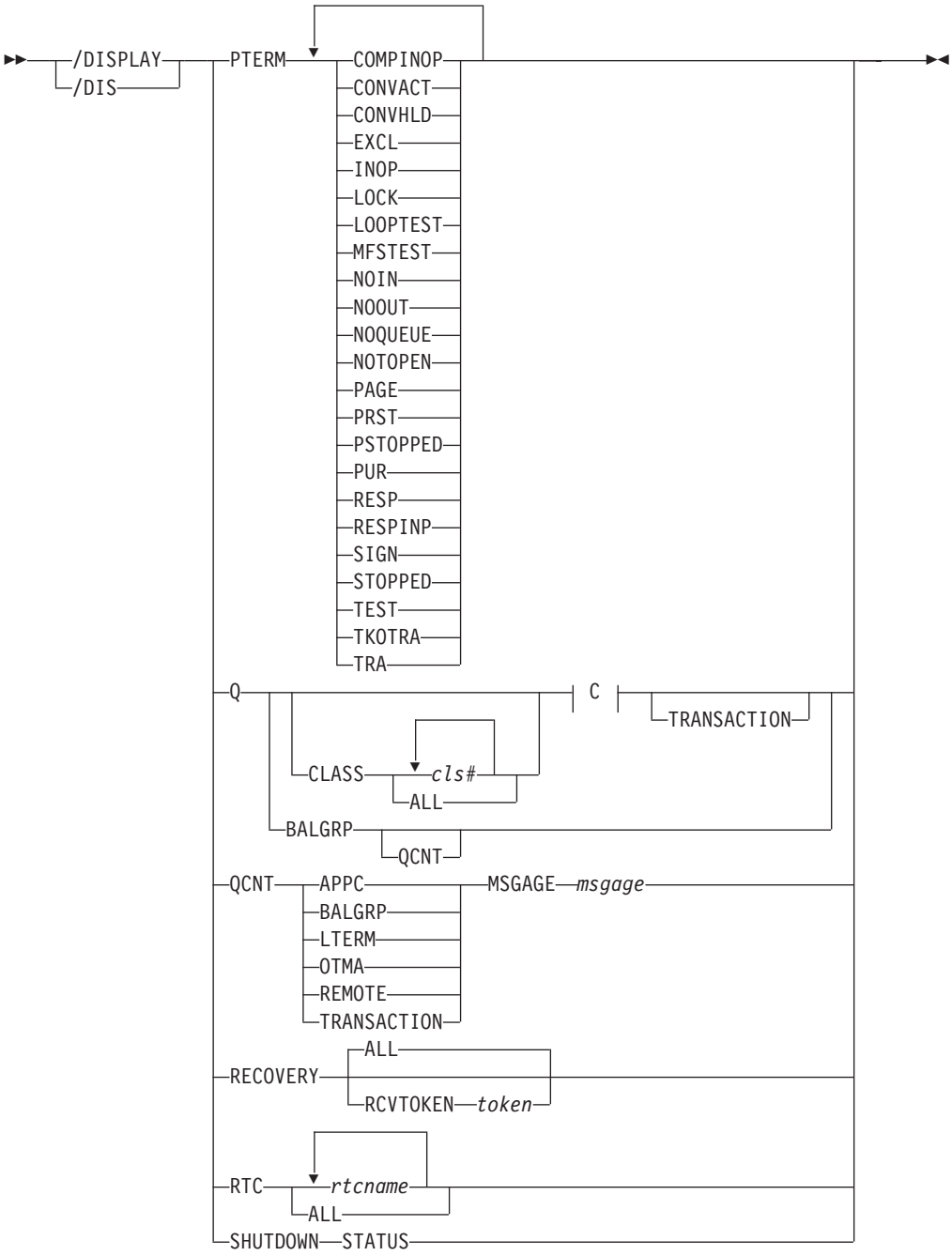
/DISPLAY Command: OASN through OVERFLOWQ



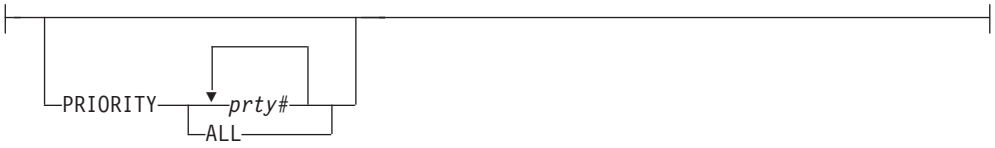
/DISPLAY Command: POOL through PSB



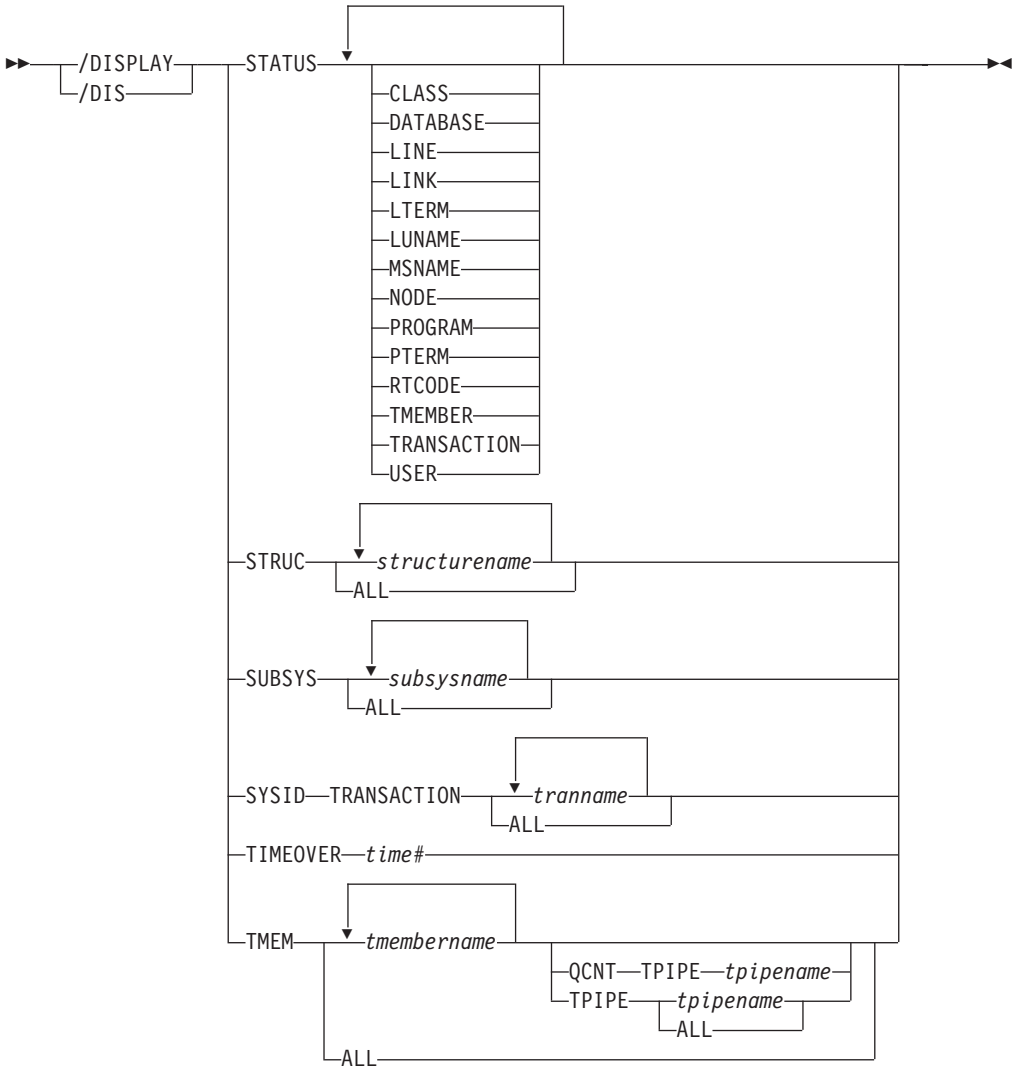
/DISPLAY Command: PTERM through SHUTDOWN STATUS



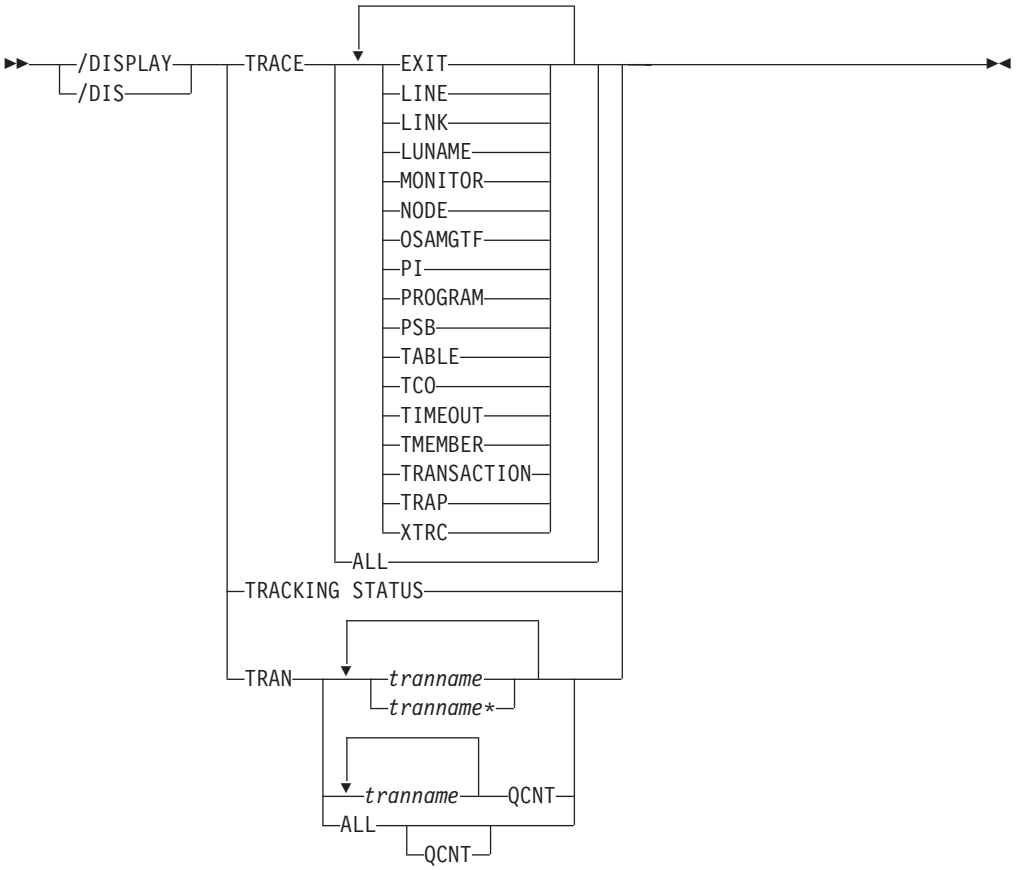
C:



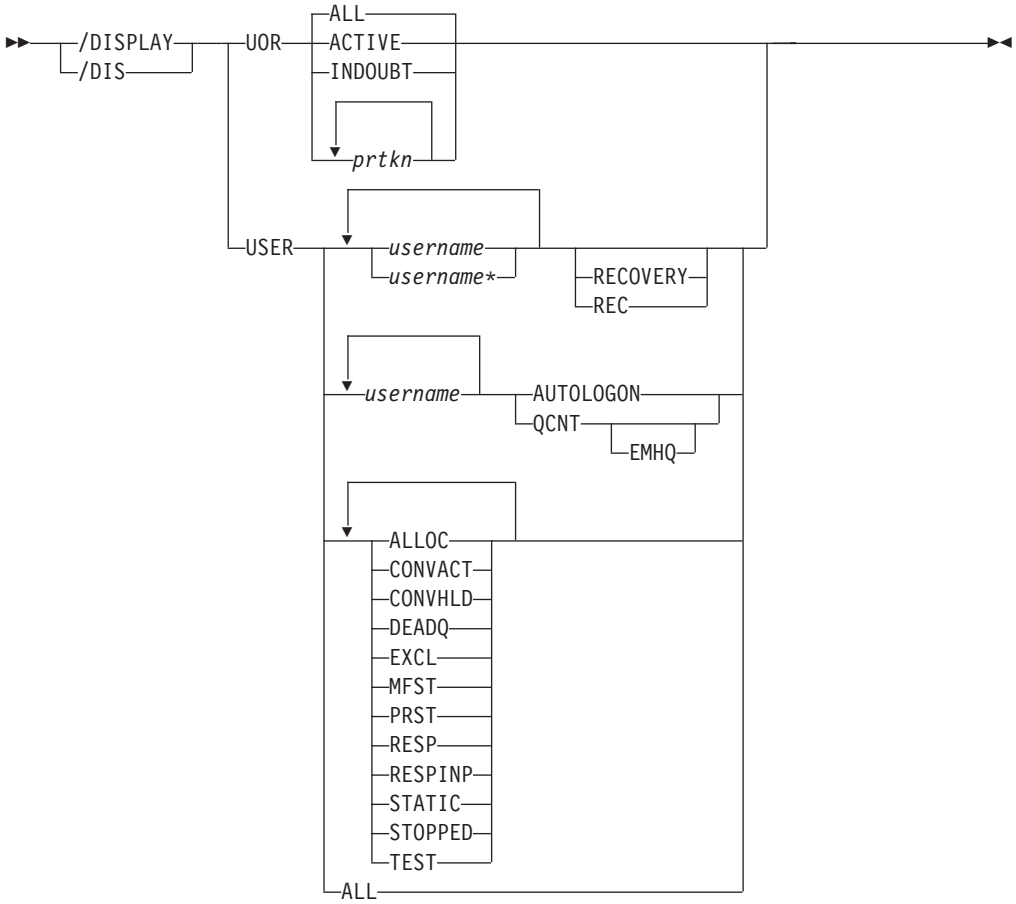
/DISPLAY Command: STATUS through TMEM



/DISPLAY Command: TRACE through TRAN



/DISPLAY Command: UOR through USER



Environments and Keywords

Table 45 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 45. Valid Environments for the /DISPLAY Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/DISPLAY	X	X	X
ACT	X	X	X
AFFIN	X		X
AOITKN	X	X	X
APPC	X		X
AREA	X	X	
ASMT	X		X
AUTOLOGON	X		X
BALGRP	X		X
BKERR	X	X	
CCTL	X	X	
CLASS	X		X
CONV	X		X
CPLOG	X	X	X
CQS	X		X
DB	X	X	
DBD	X	X	
DC	X		X
DESC	X		X
EMHQ	X		X
EXIT	X		X
FDR	X	X	
FPV	X	X	
HSB	X		X
HSSP	X	X	
INDOUBT	X	X	
INPUT	X		X
LINE	X		X
LINK	X		X
LTERM	X		X
LU	X		X
MADSIOT	X	X	
MASTER	X		X
MODE	X		X
MODIFY	X	X	X
MONITOR	X	X	X

Table 45. Valid Environments for the /DISPLAY Command and Keywords (continued)

Command / Keywords	DB/DC	DBCTL	DCCTL
MSGAGE	X		X
MSNAME	X		X
MSPLINK	X		X
NODE	X		X
OASN	X	X	X
OLDS	X	X	X
OSAMGTF	X	X	
OTMA	X		X
OUTPUT	X		X
OVERFLOWQ	X		X
PI	X	X	
PGM	X	X	X
POOL	X	X	X
PRIORITY	X		X
PSB	X	X	X
PTERM	X		X
Q	X		X
QCNT	X		X
RECOVERY	X	X	
REGION	X	X	X
REMOTE	X		X
RTC	X		X
SHUTDOWN	X	X	X
STATUS	X	X	X
STRUC	X		X
SUBSYS	X	X	X
SYSID	X		X
TABLE	X	X	X
TCO	X	X	X
TIMEOUT	X		X
TIMEOVER	X		X
TMEM	X		X
TPIPE	X		X
TPNAME	X		X
TRACE	X	X	X
TRACKING	X	X	X
TRAN	X		X
TRAP	X		X
UOR	X	X	
USER	X		X

Table 45. Valid Environments for the /DISPLAY Command and Keywords (continued)

Command / Keywords	DB/DC	DBCTL	DCCTL
XTRC	X	X	X

Usage

/DISPLAY displays the processing status of IMS message queues and processing resources. /DISPLAY can provide helpful information for controlling IMS operations.

The output from a /DISPLAY command consists of headings that describe the display, the information requested, and a time stamp showing Julian date and time, in one of the following two formats:

yyddd/hhmmss

yyyyddd/hhmmss

You control the timestamp format by specifying the YEAR4= execution parameter.

If the command includes a generic parameter that does not match any existing resource, an error message results.

See Appendix G, “Status and Attributes for the /DISPLAY Command,” on page 855 for a list of attributes and status in the /DISPLAY command.

In an IMSplex, when status is copied to RM at signoff/logoff time, the status (and control blocks if the resource is dynamic) are deleted from the local system. Pre-IMS Version 9 Automated Operator Interface (AOI) application programs that issue /DISPLAY commands will not see some status information that is kept in RM. When a /DISPLAY command is issued for NODE, USER, and LTERM resources from the command master, global status from RM may be displayed. The output of the display command will be dependent on the following guidelines:

Command Master IMS System

- Displays information from the local system
- Displays information from RM that is not owned, or is owned by a failed system

Other IMS Systems

Displays information from the local system

Use of ALL, generic parameters, the /DISPLAY STATUS command, and the /DISPLAY CONVERSATION command without a specific resource name will result in extensive accesses to the Resource Manager for global information, and their use should be carefully considered.

/DISPLAY ACTIVE

The /DISPLAY ACTIVE command displays region and DC information associated with an IMS system. The region is scheduled to an application program and the IMS resources are assigned.

When you issue /DISPLAY ACTIVE from a specific environment, such as DCCTL, only the information that is valid for that environment is displayed.

Examples of using the /DISPLAY ACTIVE command are shown in

- "Example 1 for /DISPLAY ACTIVE Command" on page 282
- "Example 2 for /DISPLAY ACTIVE Command" on page 282
- "Example 3 for /DISPLAY ACTIVE Command" on page 282
- "Example 4 for /DISPLAY ACTIVE Command" on page 283
- "Example 5 for /DISPLAY ACTIVE Command" on page 283
- "Example 6 for /DISPLAY ACTIVE Command" on page 283
- "Example 7 for /DISPLAY ACTIVE Command" on page 284
- "Example 8 for /DISPLAY ACTIVE Command" on page 284
- "Example 9 for /DISPLAY ACTIVE Command" on page 284
- "Example 10 for /DISPLAY ACTIVE Command" on page 285
- "Example 11 for /DISPLAY ACTIVE Command" on page 285
- "Example 12 for /DISPLAY ACTIVE Command" on page 285
- "Example 13 for /DISPLAY ACTIVE Command" on page 286

DC Displays only the DC subset of the output. The DC portion of the display can include:

- The status of active VTAM ACBs. If MNPS is used for XRF, then both the APPLID ACB and the MNPS ACB are displayed. If MNPS is not active, only the APPLID ACB displays.
- User Variable (USERVAR) name if using an XRF. If MNPS is active, USERVAR is invalid and is not displayed. Instead, the MNPS ACB name is displayed.
- VTAM generic resource name (GRSNAME) and its current status.
- The APPC IMS LU name, the APPC connection status, and the APPC timeout value.
- The APPC/OTMA SHARED QUEUE status, the number of RRS TCBs attached and the number of AWEs (work-elements) queued to the RRS TCBs.
- OTMA status.
- The number of active LINES (BTAM terminals) receiving input or sending output.
- The number of active NODES receiving input or sending output.
- The number of active LINKS (half sessions) receiving input or sending output.

REGION
Displays only the REGION subset of the output. The display consists of active regions.

Output for the `/DISPLAY ACTIVE DC` command is listed below:

OTMA GROUP
Status of the IMS Open Transaction Manager Access (OTMA) group. The status can be either `ACTIVE` or `NOTACTIVE`.

There can be only one OTMA group active at one time.

VTAM ACB
The status of the VTAM ACB, which can be OPEN, CLOSE PENDING, or CLOSED. The status of LOGONS can be ENABLED or DISABLED. The base IMS LU name and the current APPC connection status are shown if IMS is running with an APPC/MVS-capable operating system.

The following are displayed for the VTAM ACB:

APPLID=
The application identification name for IMS. If MNPS is used with XRF, then it is the APPLID ACB.

USERVAR=
The user name for IMS. Only applicable for XRF-capable or RSR-capable systems.

GRSNAME=
The VTAM generic resource name. Displayed as blanks if no generic resource name exists.

Restriction: IMS displays either the USERVAR or the GRSNAME. For non-XRF or non-RSR subsystems (no USERVAR specified), IMS displays the GRSNAME, which can be blank.

If the IMS system is XRF, the following fields are appended to the date/time stamp:

RSEname	Recoverable service element name
System	ACTIVE or BACKUP

MNPS=
The MNPS ACB name if MNPS is used for XRF.

STATUS
One of the following:

AWAITING SNAPQ
Alternate system awaiting SNAPQ checkpoint from active

SYNCHRONIZING
Alternate system processing SNAPQ checkpoint

TRACKING
Alternate system is in process of tracking active system

TAKEOVER REQUESTED
Alternate system requests a takeover

TAKEOVER IN PROGRESS
Alternate system in process of taking over workload from active

I/O TOLERATION
New active system in I/O toleration phase of processing prior to becoming a true active system.

Output for the `/DISPLAY ACTIVE REGION` command is listed below:

REGID
Region identifier. For DBCTL, the thread number is displayed.

JOBNAME
The name of the job processing in the region.

If no regions of that type exist, the job name is set to MSGRGN, FPRGN, BATCHRGN, or DBTRGN, depending on the region type. DBTRGN is only displayed for systems that support DBT threads.

DBTRGN, BATCHRGN, or FPRGN is displayed for DBCTL.

TYPE Type of application program processing in the region, or type of region. BMH, BMP, DBRC, DBT, DLS, and FPU are displayed for DBCTL.

The following region/program types can be displayed:

BMH HSSP processing

BMP Batch message processing

BMPE Batch message processing for an external subsystem thread

DBRC DBRC address space

DBT DBCTL thread

DLS DL/I address space

FP Fast Path

FPE Fast Path processing for an external subsystem

FPM Fast Path message-driven program

FPME Fast Path message-driven program for an external subsystem thread

FPU Fast Path utility program

FPUE Fast Path utility program for an external subsystem thread

TP Teleprocessing

TPE Message-driven program for an external subsystem thread

TPI Transaction Program Instance. A CPI Communications driven transaction is running in the region.

TRANSACTION/STEP
Transaction code being processed by region, or NONE, if there are no regions of that type.

PROGRAM
Name of the program processing in the region.

DFSCPIC is displayed for CPI Communications driven transaction programs that have not issued a DL/I APSB call to allocate a PSB.

For DBCTL, SB name is displayed.

STATUS
Status of the region, which can be one of the following:

ACTIVE-DBCMD
A /DBD or a /DBR command is in progress and waiting for the region to terminate before the /DBD or /DBR can complete.

AVAILABLE
The availability of active threads. The region is available to schedule an application.

SCHEDULED
The application program is being scheduled.

TERMINATING
The application program is being terminated.

UNAVAILABLE
An active DBT thread is unavailable. An application is using the region, even though the application is not currently scheduled. This region is therefore not available to any other application.

WAITING
The MPP region is waiting for work.

WAIT-AOI
An AO application issued a GMSG call with the WAITAOI subfunction specified, but there are no messages for the AO application to retrieve.

WAIT-BLOCKMOVER
An application control block cannot be loaded because the ACB block mover is busy.

WAIT-CMD/PENDING
A /DBDUMP, /DBRECOVERY, or /START command is in progress.

WAIT-INPUT
The application program is in WAIT-FOR-INPUT (WFI) mode.

WAIT-INTENT
The application program's intent for a database conflicts with the use of the database by a scheduled program.

WAIT-INTENT/POOL
Indicates either the application program's intent for a database conflicts with the use of the database by a scheduled program, or a temporary shortage of DMB, PSB, or PSB work area pool space exists.

WAIT-INTENT SCHD
The IMS transaction scheduler detected an application scheduling intent failure. (For example, Load Balancing.)

WAIT-I/O PREVEN
A BMP region which accesses a GSAM database cannot schedule until I/O prevention has completed.

WAIT-MESSAGE
The application program is in a pseudo WAIT-FOR-INPUT (WFI) mode. The application is scheduled and is waiting for a message.

WAIT-POOLSPACE
A temporary shortage of DMB, PSB, or PSB work area pool space exists.

WAIT-SWITCHOVER
The alternate system is tracking the active system.

WAIT-SYNCPOINT
The application in the region is now in SYNC POINT.

WAIT-EPCB POOL
A temporary shortage of EPCB pool space exists.

WAIT-RRS PC
The application program has a protected conversation with an OTMA client that is processing a sync point. Sync point can continue after the OTMA client issues either an SRRCMIT or SRRBACK call. Or, the application program is part of a cascaded family and is processing a sync point. APPC/OTMA SMQ Enablement uses RRS cascaded transaction support to synchronize the back-end and the front-end system.

CLASS
One of the classes associated with the region. The region can have from 1 to 4 classes, whose values range from 1 to 999.

Note: If /DISPLAY ACTIVE command is issued, the output for both DC and REGION is displayed.

/DISPLAY AFFINITY

The /DISPLAY AFFINITY command displays current affinities the IMS subsystem has with specific nodes.

This command is only valid when you use VTAM generic resources.

An example of using the /DISPLAY AFFINITY command is shown in “Example for /DISPLAY AFFINITY Command” on page 286

/DISPLAY AOITOKEN

/DISPLAY AOITOKEN displays all AOI tokens in the system. An example using /DISPLAY AOITOKEN is shown in “Example for /DISPLAY AOITOKEN Command” on page 286

The following information is displayed for each AOI token:

AOITOKEN
AOI token name.

ENQCT
Total number of messages enqueued. In a shared-queues environment, only shows messages enqueued for the local subsystem.

QCT Number of messages currently queued to the token and not yet retrieved by the AO application. In a shared-queues environment, only shows messages enqueued for the local subsystem.

SEGS Number of segments in messages queued (QCT) to the AOI token.

W-REGID
The region IDs of AO applications that issued a GMSG call with the WAITAOI subfunction specified. This AO application is in a wait state, since there are currently no messages for it to retrieve.

If QCT=0 and W-REGID=NONE, then the AOI token is deleted at the next checkpoint.

/DISPLAY APPC

APPC displays inquiries about LU 6.2 related activities, including the current outbound LU if it is different from the base LU.

The output for /DISPLAY APPC is as follows:

IMSLU
The IMSLU is a base IMS LU name. This LU is always a network-qualified LU name. It is displayed as N/A.N/A if IMS is not connected to APPC/MVS. For example, if status is DISABLED, FAILED, or STARTING, N/A.N/A applies. For the CANCEL status, the field can contain either the LU name or N/A.N/A.

#APPC-CONV
LU 6.2 inbound and outbound conversation count.

SECURITY
RACF security level, which is one of the following:

- CHECK
- FULL
- NONE
- PROFILE

STATUS
Current APPC connection status. The possible values for APPC connection status are:

CANCEL
Shown after the /STOP APPC CANCEL command.

DISABLED
Shown when APPC/IMS is not identified or connected to APPC/MVS.

ENABLED
Shown after successful /START APPC command completion.

FAILED
Shown after unsuccessful /START APPC command.

OUTBOUND
Shown on the XRF alternate system as it tracks the active. It is also shown on the active if IMS is unable to initiate normal APPC processing. For example, the /START APPC command can be reissued to attempt APPC enablement.

PURGING
Shown after the /PURGE APPC command.

STARTING
Shown after the /START APPC is issued and before the status is set to ENABLED or FAILED.

STOPPED
Shown after the /STOP APPC command.

DESIRED
Desired APPC connection status. This status is changed by the /PURGE, /START and /STOP APPC (CANCEL) commands. The possible values for desired APPC connection status are:

CANCEL
Shown after /STOP APPC CANCEL

DISABLED
Shown when APPC/IMS is not identified or connected to the APPC/MVS

ENABLED
Shown after /START APPC

OUTBOUND
Shown on the XRF alternate system as it tracks the active

PURGING
Shown after /PURGE APPC

STOPPED
Shown after /STOP APPC

GRNAME
The IMS/APPC generic LU name (if VTAM Generic Resources is activated).

TYPE BASE for BASE LU or OUTB for OUTBOUND LU.

/DISPLAY AREA

/DISPLAY AREA displays data sets, status conditions, control intervals, and databases associated with an area.

Examples of using the /DISPLAY AREA command are shown in:

- "Example 1 for /DISPLAY AREA Command" on page 287
- "Example 2 for /DISPLAY AREA Command" on page 287
- "Example 3 for /DISPLAY AREA Command" on page 288
- "Example 4 for /DISPLAY AREA" on page 288

The output for /DISPLAY AREA includes:

AREANAME
Name of the Fast Path DEDB area.

DDNAME
Names of the data sets that contain that area.

CONDITIONS
Status conditions associated with the area or ADS.

Status conditions associated with the area include:

IC HSSP image copy is currently active on the area.

MADSIOT
Displays only the areas that are currently in a long busy condition or are in long busy recovery mode.

NOT-OPEN
The area has yet to be opened. This will be done when first call is issued to the area.

RECALL
Area is in recall.

RECOVERY-NEEDED
The area has been marked RECOVERY-NEEDED. The database recovery utility needs to be run to restore this area.

STOPPED
The area is in STOPPED status. All access to it is prohibited. This can be changed with the /START command.

Status conditions associated with the ADS include the following:

COPY-PHASE
The CREATE utility is currently in the COPY phase on this ADS. The CREATE utility must complete before anything can be done with this ADS.

FORMAT-PHASE
The CREATE utility is currently in the FORMAT phase on this ADS. The CREATE utility must complete before anything can be done with this ADS.

PRE-OPEN FAILED
XRF PREOPEN was not successful for this ADS.

RECALL
Area is in Recall.

SEVERE-ERROR
This ADS has had a severe I/O error (write error to second CI, in which Fast Path keeps its control information).

UNAVAILABLE
This ADS has been marked unavailable, probably due to I/O errors. (EQE REMAIN = 0).

EQECT
Specifies the count of I/O errors for the area.

TOTAL SEQ DEPENDENT
Total control intervals defined for sequential dependent space.

UNUSED SEQ DEPENDENT
Unused control intervals for sequential dependent space. See Chapter 8: Database Design Considerations for Fast Path in the *IMS Version 9: Administration Guide: Database Manager* for more information about sequential dependent space management.

TOTAL DIRECT ADDRESSABLE
Total control intervals defined for direct addressable space.

UNUSED DIRECT ADDRESSABLE
Unused control intervals for direct addressable space.

DBNAME
Database name.

EEQECT
Count of write error EEQE for this area.

IOVF Independent overflow area statistics.

These statistics are only displayed if `IOVF` is specified on the `/DIS AREA` command (`/DIS AREA ALL IOVF` or `/DIS AREA areaname IOVF`). If the IOVF parameter is not specified on the `/DIS AREA` command, `N/A` appears in the display output.

If the `/DIS AREA IOVF` command is entered on an RSR tracking subsystem, the IOVF statistics might be incorrect.

When the `ALL` parameter is used with the `IOVF` keyword (for example, `/DIS AREA ALL IOVF`), performance can be affected depending on the number and size of areas involved. If large areas, large numbers of areas, or both are involved, the control region can appear to be stopped while processing this command.

“Attribute” is a reserved parameter. Attributes listed with Table 46 on page 220 can be used with the AREA keyword. Areas are selected for display based on the attribute or attributes specified. For example, `/DISPLAY AREA STOPPED` will only display areas that are currently stopped. The attribute display format is the same as

the standard AREA display. Attributes usually correspond to the conditions displayed for areas. Any exceptions are flagged in Table 46

The attributes are reserved parameters for the /DISPLAY AREA command and cannot be used to name areas. In the following table, listed are the attributes that can be used with the AREA keyword.

Table 46. /DISPLAY AREA Command Attributes

IC	NOTOPEN[1]
OFR	RECOVERN[2]
RECALL	STOPPED

Note:

1. Corresponds to AREA condition NOT OPEN.
2. Corresponds to AREA condition RECOVERY NEEDED.

2 If an IMS Fast Path utility is active when the /DISPLAY AREA command is issued, the
2 following output line is displayed:
2
```
UTILITY XXXX   PVTTOT#   YY  PVTAVL#  ZZ   IMSID:SSSS   TTTT
```

2 The output values represent:
2 • XXXX - the active utility name or VSO function
2 • YY - the total number of private buffers available to the utility
2 • ZZ - the number of unused private buffers available to the utility
2 • SSSS - the IMSID of the IMS system on which the utility is active
2 • TTTT - the IMSID of the IMS system on which the display was issued

| There are six IMS Fast Path utilities, two VSO functions, and one IBM DBTOOL
| utility that can be displayed. They include:
| • HSSP (High Speed Sequential Processing utility)
| • HSRE (DBFUHDR0, High Speed Reorganization utility)
| • MDL0 (DBFUMDL0, SDEP Delete utility)
| • MSC0 (DBFUMSC0, SDEP Scan utility)
| • MRI0 (DBFUMRI0, DEDB Create utility)
| • MMH0 (DBFUMMH0, DEDB Compare utility)
| • PRLD (VSO Preload is active)
| • VXPL (Shared VSO Preload is active)
| • TOOL (an IBM DBTOOL Online utility)

/DISPLAY ASSIGNMENT

/DISPLAY ASSIGNMENT displays resource assignments.

Examples of using the /DISPLAY ASSIGNMENT command are shown in:

• “Example 1 for /DISPLAY ASSIGNMENT Command” on page 288
• “Example 2 for /DISPLAY ASSIGNMENT Command” on page 293

The following information is displayed when this command is used:

LINE PTERM
The logical terminal names associated with the specified line and physical terminal.

LINK The physical link, SYSIDs, and logical link paths assigned to the specified logical link.

LTERM
The communication line and physical terminal or node and component associated with the specified logical terminal name. When using ISC or a dynamic terminal, the displayed output for a logical terminal assigned to the VTAM pool shows the assigned node, if any, and the user name. The LTERM parameter can be generic.

MSNAME
The physical link and logical link assigned to the specified logical link path.

MSPLINK
The type, address, and logical link associated with the specified physical link. For MSC with VTAM, MSPLINK also displays the maximum number of allowed sessions and VTAM node name.

NODE The logical terminal names associated with the specified node. When using ISC, the displayed output shows at least one line of information for each half-session of the node, shows whether the half-session is active, and shows whether LTERMs are assigned. If the half-session is active, the user identifier is displayed. If no LTERMs are assigned, NONE is indicated. The USER keyword is valid for ISC nodes and non-ISC nodes with signed-on users. The NODE parameter can be generic.

SYSID The physical link, logical link, and logical link path assignments associated with the specified system identification.

USER All the USER structures and USERIDs that match the parameter specified. These are the following:

- The USER field shows the USER structure name. N/A appears for all static USERIDs signed on to static nodes.
- The USERID field displays the RACF USERID that was used to sign on to the node. N/A appears if a USER structure is not associated with a node. In this case, the USER structure exists to hold status or has messages inserted to it. If no messages or status exist, the USER structure is deleted at the next checkpoint.
- The node name and logical terminal names are displayed if the user is allocated to a node or signed on to a node.

The USER parameter can be generic.

/DISPLAY CCTL

`/DISPLAY CCTL` displays all coordinator controllers connected to DBCTL along with the pseudo-recovery token, recovery token, region ID, psbname, and status. Coordinator controllers that are attached to IMS show a status of ATTACHED.

Restriction The `/DISPLAY CCTL` command does not support the ODBA interface to IMS DB. Use the `/DISPLAY UOR` command for displaying status information about units of work for protected resources that are managed by RRS z/OS.

Examples of using the `/DISPLAY CCTL` command are shown in:

- "Example 1 for /DISPLAY CCTL Command" on page 293
- "Example 2 for /DISPLAY CCTL Command" on page 294
- "Example 3 for /DISPLAY CCTL Command" on page 294
- "Example 4 for /DISPLAY CCTL Command" on page 294
- "Example 5 for /DISPLAY CCTL Command" on page 294

The status of active threads can be any of the following:

ACTIVE
The region is available for scheduling an application program. ACTIVE displays all active threads.

ACTIVE-DBCMD
A /DBD or a /DBR command is in progress and waiting for the region to terminate before the /DBD or /DBR can complete.

AVAILABLE
Active threads are available. The region is available to schedule an application.

TERM The thread is terminating. The application program is being terminated.

UNAVAIL
Active threads are unavailable. An application is using the region, even though the application is not currently scheduled. This region is therefore not available to any other application.

W-BLKMVR
The thread is waiting for a block mover. An application control block cannot be loaded because the ACB block mover is busy.

W-INTENT
The thread is waiting for intent. The application program's intent for a database conflicts with the use of the database by a scheduled program.

W-POOLSP
The thread is waiting for pool space. A temporary shortage of DMB, PSB, or PSB work area pool space exists.

W-SWITCH
The thread is waiting for switch-over. The alternate system is tracing the active system.

INDOUBT displays all in-doubt threads. The status of all in-doubt threads is always displayed as INDOUBT.

/DISPLAY CONVERSATION

/DISPLAY CONVERSATION displays all BUSY and HELD conversations in the IMS system if no other keywords or parameters are specified. The command displays a 4-byte conversation ID. The conversation IDs are unique only for each user and terminal.

An example of using the /DISPLAY CONVERSATION command is shown in "Example for /DISPLAY CONVERSATION Command" on page 294

The following information is displayed when this command is used:

BUSY The conversations currently active for the system or active for a specific node, line and physical terminal, or user.

HELD The conversations currently held in the system or held from a specific node, line and physical terminal, or user.

When ISC is used with parallel sessions, one line of information for each session in a conversation is displayed. The user name is also displayed.

When the NODE keyword is used and the node is an ISC node, information for all applicable sessions of the parallel session node is displayed. For dynamic terminals, the conversations of the currently associated user are displayed. The NODE parameter can be generic.

When the USER keyword is specified, all conversations that apply to that user are displayed. The USER parameter can be generic. In an IMSplex, use of the /DISPLAY CONVERSATION command without a specific resource name will result in extensive accesses to the Resource Manager for global information, and its use should be carefully considered.

For an IMS conversational transaction issued from an LU 6.2 application program, /DISPLAY CONVERSATION shows the LU name and user ID (if available) of the LU 6.2 conversation along with the IMS conversation ID and status.

/DISPLAY CPLOG

/DISPLAY CPLOG displays the value of the IMS execution parameter, CPLOG. CPLOG specifies the number of system log records between system-generated checkpoints.

/DISPLAY CQS

/DISPLAY CQS displays information about the Common Queue Server (CQS), including the CQS job name, version number, and current status. Valid status conditions are CONNECTED and DISCONNECTED.

This command is only valid in a shared-queues environment.

An example of using the /DISPLAY CQS command is shown in “Example for /DISPLAY CQS Command” on page 296

/DISPLAY DATABASE

/DISPLAY DATABASE displays the status of specified databases. If a specified database is a DEDB, the associated DEDB areas are also displayed.

The /DISPLAY DATABASE command can be used on HALDBs. For more information see Appendix H, “High Availability Large Database Commands,” on page 867

An example of using the /DISPLAY DATABASE command is shown in “Example 1 for /DISPLAY DATABASE Command” on page 296

The following information is displayed:

DATABASE
Name of the database.

TYPE Type of database, which can be one of the following:

- DL/I

- DEDB
- MSNR (refers to MSDB non-related databases)
- MSRD (refers to MSDB related dynamic databases)
- MSRF (refers to MSDB related fixed databases)
- PHIDAM
- PHDAM
- PSINDEX
- PART
- blank (if the database status is NOTINIT)

TOTAL SEQ DEPEND
Total control intervals defined for sequential dependent space, which applies to areas associated with DEDBs. This column only appears if Fast Path is installed.

UNUSED SEQ DEPEND
Unused control intervals for sequential dependent space. This column only appears if Fast Path is installed.

TOTAL DIRECT ADDRESS
Total control intervals for independent overflow part of direct addressable space, which applies to areas associated with DEDBs. This column only appears if Fast Path is installed.

UNUSED DIRECT ADDRESS
Total unused control intervals for independent overflow part of a direct addressable space, which applies to areas associated with DEDBs. This column appears only if Fast Path is installed. This IOVF count is updated on a timed basis, with a default timer of 2 hours. To customize this IOVF counter timer, see the IOVFI parameter as described in *IMS Version 9: Installation Volume 2: System Definition and Tailoring*.

ACC Type of access to the database, which can be one of the following:

RO	Read only
RD	Read
UP	Update
EX	Exclusive

CONDITIONS
Status conditions of the database, which can be one or more of the following:

- ALLOCF
- ALLOCS
- BACKOUT
- EEQE
- INQONLY
- NOTINIT
- NOTOPEN
- OFR
- OLR
- RECALL
- RECOVERY

- RNL
- STOPPED
- LOCK

Note: The OFR parameter is only valid when issued from an RSR tracking subsystem.

"Attribute" is a reserved parameter. Attributes listed with Table 47 can be used with the DATABASE keyword. Databases are selected for display based on the attribute or attributes specified. For example, `/DISPLAY DATABASE OFR` will only display those databases that are currently being recovered by online forward recovery (OFR). The attribute display format is the same as the standard DATABASE display. Attributes usually correspond to the conditions displayed for databases.

3 **Restriction**: The OLR attribute cannot be entered with any other attribute.

Table 47. /DISPLAY DATABASE Command Attributes

ALLOCF	ALLOCS
BACKOUT	EEQE
INQONLY	LOCK
NOTINIT	NOTOPEN
3 OFR	OLR
RECALL	STOPPED

Attributes are reserved parameters for the `/DISPLAY DATABASE` command and cannot be used to name databases.

/DISPLAY DATABASE with BKERR

`/DISPLAY DATABASE` with BKERR displays error queue elements associated with a DL/I database and displays incomplete backouts.

The output includes:

DATABASE
Name of the database with error queue elements or backout pending data.

ERROR DD
DDNAME of the data set with the error queue element

TYPE Type of error queue element, which is one of the following:

BACK PSB	PSB needing backout
BLOCK	OSAM block number or VSAM CI RBA
DATE	Date
IDT	In-doubt
IDT+WRT	In-doubt and write error
IDT+IOT	In-doubt and I/O toleration
IOT	I/O toleration
IOT/VSO	I/O toleration for VSO areas
PRM	DBRC permanent
RD	Read error

TIME	Time
USE	DBRC user created
WRT	Write error

The EEQE information might not be current if the database is registered with DBRC and is not open, because IMS updates EEQE information from DBRC when the database is opened.

The /DIS DB command with BKERR does not display individual RBAs for I/O tolerated non-SDEP CIs of a DEDB area defined with the Virtual Storage Option (VSO). Instead, a single entry with a value of 00000001 is displayed; the EEQE type is set to IOT/VSO. I/O tolerated SDEP CIs for the area are displayed individually by RBA. See“Example 4 for /DISPLAY DATABASE Command” on page 297.

/DISPLAY DBD

/DISPLAY DBD displays, for databases that are being accessed, their type, the PSBs accessing them, and the type of access. /DISPLAY DBD can be used only if Fast Path is installed.

An example of using /DISPLAY DBD is shown in “Example for /DISPLAY DBD Command” on page 299

/DISPLAY DESCRIPTOR

/DISPLAY DESCRIPTOR displays LU 6.2 descriptors. All specified values are displayed, as are defaults/blanks for values not specified in the descriptor.

An example of using /DISPLAY DESCRIPTOR is shown in “Example for /DISPLAY DESCRIPTOR Command” on page 299

The output for /DISPLAY DESCRIPTOR is as follows:

DESC Name of the descriptor.

LUNAME
LU name value in the descriptor. The network-qualified LU name is optional for the LUNAME keyword.

MODE VTAM LOGON mode table entry value in the descriptor.

SIDE APPC z/OS side table entry value in the descriptor.

SYNCLEVEL
APPC synchronous level in the descriptor; either NONE or CONFIRM.

TPNAME
TP name value in the descriptor. If the tpname is too long, a + is appended at the end, and the rest is continued on a second line.

TYPE APPC conversation type value in the descriptor; it is either BASIC or MAPPED.

/DISPLAY FDR

/DISPLAY FDR displays the current status for IMS Fast Database Recovery regions.

IMS rejects this command if the active IMS subsystem is not connected to a Fast Database Recovery region.

An example of using /DISPLAY FDR is shown in “Example for /DISPLAY FDR Command” on page 299

FDR-REGION
Status of the Fast Database Recovery region to which the IMS subsystem is connected. The status can be ACTIVE or NO_ACTIVE.

GROUPNAME
The XCF group name specified in the DFSFDRxx member of IMS.PROCLIB.

TIMEOUT
The failure-detection timeout value specified in the DFSFDRxx member or set by a /CHANGE FDR command.

/DISPLAY FPVIRTUAL

/DISPLAY FPVIRTUAL displays the areas loaded into a z/OS data space or coupling facility. That is, those areas that have been specified with the virtual storage option (VSO) are displayed with this command.

Examples of using /DISPLAY FPVIRTUAL are shown in “Example for /DISPLAY FPVIRTUAL Command” on page 300

Each data space is displayed, along with each of the areas that reside in the data space. All numbers shown are in decimal values.

DATASPACE
Data space number.

MAXSIZE(4K)
Maximum available size, in 4 KB blocks. This is the amount of space available in the data space for user data.

AREANAME
Name of the area located in the data space or in the coupling facility.

AREASIZE(4K)
Actual storage available for a particular area, in 4KB blocks.

The actual space usage can be significantly less than what is reserved if a VSO dataspace is not preloaded.

OPTION
A data space option (which can be DREF) or an area option (which can be PREL or PREO). Area options are defined to DBRC with the INIT.DBDS or CHANGE.DBDS command.

STRUCTURE
Name of the coupling facility structure allocated for this area.

ENTRIES
Number of area CIs in the structure.

CHANGED
Number of area CIs in the structure that have been updated. IMS writes these updated CIs to DASD (and resets this value) when you enter any of the following commands:

- /CHECKPOINT

- /STOP AREA
- /VUNLOAD

2 **AREACI#**
2 Total number of area CIs that can be loaded into the structure.

POOLNAME
Name of the private buffer pool used by the area.

/DISPLAY HSB

/DISPLAY HSB displays the system-related information in an XRF environment. If the command is entered on the active system, information about the active system is displayed. If the command is entered on the alternate system, information about both systems is displayed.

An example of using the /DISPLAY HSB command is shown in "Example for /DISPLAY HSB Command" on page 301

The output for /DISPLAY HSB includes:

RSENAME
Recoverable service element name.

STATUS
System entering command (ACTIVE or BACKUP).

MODETBL
Indication of whether (IOT) or not (blank) the system is in I/O toleration mode. This field is displayed only if the command is entered on the active system.

PHASE
Displays only if the command is entered on the alternate system; indicates the phase of the alternate system. The following phases can be displayed:

INIT	Initialization phase
SYN	Processing SNAPQ checkpoint
TRK	Tracking active system
TKO	Taking over workload from active

IMS-ID
IMS identifier of system entering command.

VTAM UVAR
VTAM generic name.

ACT-ID
IMS identifier of active system; field is displayed only if the command is entered on the alternate system.

LOG-TIME
Time associated with the log record currently being read by the alternate system in the process of tracking the active system. The length of time the alternate system lags behind the active system in reading the log is the difference between the current time, as shown by the timestamp, and the log time. Some of this time difference might be due to differences in the synchronization of the clocks of the processors.

SURVEILLANCE
Indicates one of the following surveillance mechanisms:

LNK IMS ISC link

LOG IMS system log

RDS IMS restart data set

INTERVAL
The surveillance interval, in seconds, for the specific surveillance mechanism

TIMEOUT
The surveillance timeout value, in seconds, for the specific surveillance mechanism

STATUS
The status of the specific surveillance mechanism (ACTIVE or INACTIVE).

TAKEOVER CONDITIONS
The following takeover conditions may apply:

ALARM
The alarm on the alternate processor will be sounded when takeover is requested.

AUTO Automatic takeover is in effect.

Following the display line of ALARM, AUTO, or both, the takeover conditions specified on the SWITCH control statement of the `DFSHSBxx PROCLIB member` are shown. (xx is the SUF= parameter of the IMS procedure.) There are two groups per line and each group is separated by an asterisk. Each group can include one or more of the following:

VTAM VTAM TPEND exit

LNK IMS ISC link

LOG IMS system log

RDS IMS restart data set

/DISPLAY HSSP

`/DISPLAY HSSP` displays information about the current HSSP (high-speed sequential processing) activities.

An example of using the `/DISPLAY HSSP` command is shown in “Example for /DISPLAY HSSP Command” on page 302

The output for `/DISPLAY HSSP` includes:

RGN Region ID as displayed in the `/DISPLAY ACTIVE` command (three digit PST number)

TYP Region type (BMP only)

JOBNAME
Job name

PSB Program Specification Block name

DB Database name

AREAS-OPTIONS
Information on the SETO statement:

AREA Area name

PCB Information on whether the named area has updated intent.

PROCOPT
Whether PROCOPT H is specified.

IC Image Copy in process.

UPD Information on whether the area is updated, which is one of the following:

N The PCB has no update intent on the named area.

Y The PCB has update intent on the named area.

OPTION
Information on the execution

IC Image Copy in process

/DISPLAY LINE

`/DISPLAY LINE`, when followed by a valid line number or ALL, displays the status and queue counts for the specified communication line or line and physical terminal. The queue count for `/DISPLAY LINE` is reset to zero after an IMS cold start.

Lines are selected for display based on the attributes specified. Attributes that can be used with the LINE keyword are shown in Table 48 The attribute display format is the same as the standard LINE display. The attributes correspond to the conditions displayed for lines. For example, `/DISPLAY LINE LOOPTEST` displays all lines that are currently in looptest mode.

Table 48. /DISPLAY LINE Command Attributes

DEACT	NOQUEUE	RESPINP
IDLE	NOTOPEN	STOPPED
LOOPTEST	PSTOPPED	TKOTRA
NOIN	PUR	TRA
NOOUT	RESP	

An example of using the `/DISPLAY LINE` command is shown in “Example for /DISPLAY LINE Command” on page 302

QCNT
Specifies that global queue count information is to be displayed. If `QCNT` is not specified, local queue counts are displayed; when it is specified, no local queue counts are displayed.

This keyword is only valid in a shared-queues environment.

/DISPLAY LINK

`/DISPLAY LINK`, when followed by a valid link number or ALL, displays the status and queue counts for the specified logical link. The queue count for `/DISPLAY LINK` is reset to zero after an IMS cold start.

In addition, /DISPLAY LINK shows if the logical link is assigned to a physical link and the partner identification that is assigned to it during system definition.

The /DISPLAY LINK link# MODE command, when entered from the primary IMS system, displays in the ACT MODETBL field the mode table name from the /RESTART command or LOGON exit. The same field does not display on the secondary IMS system.

Note: N/A is displayed by /DISPLAY LINK ALL MODE for non-VTAM MSC links.

Examples for using the /DISPLAY LINK command are shown in:

- "Example 1 for /DISPLAY LINK Command" on page 303
- "Example 2 for /DISPLAY LINK Command" on page 303

MODE Displays the mode table names associated with the links specified. The output includes:

LINK Logical link number.

PARTNER
The identification specified during IMS system definition for this multiple system partnership.

DEF MODETBL
Default mode table name set by system definition or /CHANGE command. This name can be overridden with the /RSTART command or, for non-IMS session initiations, the LOGON exit.

ACT MODETBL
Mode table name actually used to initiate the session. This name is only displayed while the session is active. The field is blank at normal session termination.

QCNT
Specifies that global queue count information is to be displayed. If QCNT is not specified, local queue counts are displayed; when it is specified, no local queue counts are displayed.

This keyword is only valid in a shared-queues environment.

/DISPLAY LTERM

/DISPLAY LTERM displays status and queue counts for the specified logical terminal name. The status displayed can be one of the following:

- LOCK
- PSTOP
- PUR
- QERR
- QLOCK
- STATIC
- STOP

The LTERM parameter can be generic where the generic parameter specifies lterms that already exist.

/DISPLAY LTERM ALL does not display lterms in alphabetical order.

Examples of using the /DISPLAY LTERM command are shown in:

- "Example 1 for /DISPLAY LTERM Command" on page 304
- "Example 2 for /DISPLAY LTERM Command" on page 304

EMHQ
Specifies that global queue count information on the Expedited Message Handler Queues (EMHQ) displays when both EMHQ and QCNT are specified. If EMHQ is specified, QCNT must also be specified. This keyword is only valid in a shared-queues environment.

QCNT
Specifies that global queue count information displays. If QCNT is not specified, local queue counts are displayed; when it is specified, no local queue counts are displayed. If QCNT is specified and EMHQ is not, global queue count information on the shared message queues (MSGQ) displays.

IMS does not check if the specified name is a valid LTERM in the IMS subsystem on which the command is entered.

The /DISPLAY LTERM ALL QCNT command displays all of the LTERMs on the shared queues with a global queue count.

This command does not display queue counts for Fast Path output messages for the NODE unless EMHQ and QCNT are specified.

This keyword is only valid in a shared-queues environment.

For IMS subsystems in a shared-queues environment with the time control option, TCO, enabled, AFFINITY has been added to the queue names of TCO LTERMs. When the /DISPLAY LTERM DFSTCFI QCNT command is issued, the queue count of messages for the DFSTCFI LTERM pertains to the TCO LTERM on the local IMS subsystem. The queue count is zero priot to adding AFFINITY to the queue names.

In a shared queues environment, there are certain situations that cause an LTERM message to be retrieved from the global queue and moved to the local queue, but the message has not been delivered. If this occurs, the global queue count is zero and you need to display the local queue to see the output message.

/DISPLAY LUNAME

/DISPLAY LUNAME displays information about a specific LU 6.2 application program. After restart and any checkpoint, only LU 6.2 resources with status or messages queued are displayed.

QCNT
Specifies that global queue count information displays. Local queue counts do not display. If QCNT is not specified, local queue counts display.

This keyword is only valid in a shared-queues environment.

/DISPLAY LUNAME INPUT

/DISPLAY LUNAME INPUT displays the count of LU 6.2 inbound conversations and synchronous outbound activities for the specified LU names and their status. The status displayed can be none, stopped (STO), traced (TRA), or both stopped and traced.

/DISPLAY LUNAME INPUT can indicate that a command such as /STO LUNAME ALL INPUT or /TRACE SET ON LUNAME ALL INPUT was entered before the display that applies to all future LU 6.2 inbound conversations. The /DISPLAY LUNAME INPUT command includes:

LUNAME
LU name of the LU 6.2 application program

#APPC-CONV
The number of LU 6.2 inbound conversations

Status can be stopped (STO), traced (TRA), or stopped and traced.

/DISPLAY LUNAME OUTPUT

/DISPLAY LUNAME OUTPUT displays the count of asynchronous output messages that are being enqueued and dequeued for the LU name and the count of LU 6.2 asynchronous outbound conversations for that LU name.

/DISPLAY LUNAME OUTPUT can indicate that a command such as /STO LUNAME ALL OUTPUT or /TRACE SET ON LUNAME ALL OUTPUT was entered that applies to all future LU 6.2 outbound conversations. The status displayed can be none, stopped (STO), traced (TRA), or both stopped and traced. /DISPLAY LUNAME OUTPUT includes:

LUNAME
LU name of the LU 6.2 application program.

ENQCT
Total number of messages enqueued on this LU name. In a shared-queues environment, only shows messages enqueued for the local subsystem.

DEQCT
Total number of messages dequeued from this LU name. In a shared-queues environment, only shows messages dequeued for the local subsystem.

QCT Total number of messages still in the queue. In a shared-queues environment, only shows messages enqueued for the local subsystem.

CONVCT
Number of LU 6.2 asynchronous outbound conversations.

Status can be stopped (STO), traced (TRA), or stopped and traced.

/DISPLAY LUNAME TPNAME

/DISPLAY LUNAME TPNAME displays the number of asynchronous output messages being enqueued and dequeued and their status. The status displayed can be none, stopped (STO), traced (TRA), or both stopped and traced.

If the LU name includes a network identifier, you must specify the network-qualified LU name when using the TPNAME keyword. A network-qualified LU name refers to a different resource than a non-qualified LU name.

/DISPLAY LUNAME TPNAME includes the following:

LUNAME TPNAME
LU name and TP name of the LU 6.2 program. The first line displays the LU name and the second line displays the TP name, which is preceded by a dash. If the TP name is too long, a + sign is appended at the end, and the remainder is continued on the next line.

ENQCT
The total number of messages enqueued to this particular LU 6.2 application program. In a shared-queues environment, only shows messages enqueued for the local subsystem. If the ENQCT and DEQCT are the same and there are no messages enqueued or dequeued from that point on, these counts will be set to zero after restart or takeover.

DEQCT
The total number of messages dequeued from this particular LU 6.2 application program. In a shared-queues environment, only shows messages dequeued for the local subsystem. If the ENQCT and DEQCT are the same and there are no messages enqueued or dequeued from that point on, these counts will be set to zero after restart or takeover.

QCT The total number of messages still in the queue. Status of stopped (STO), traced (TRA), or stopped and traced. In a shared-queues environment, only shows messages enqueued for the local subsystem.

/DISPLAY MASTER

`/DISPLAY MASTER` displays the logical terminal name and the line and physical terminal number associated with the master terminal.

If the 3270 master terminal capability was generated during IMS system definition, the logical terminal name, line, and physical terminal number of the secondary master terminal are also displayed.

MASTER is not valid for use in a DBCTL environment.

An example of using the `/DISPLAY MASTER` command is shown in “Example for /DISPLAY MASTER Command” on page 307.

/DISPLAY MODIFY

The `/DISPLAY MODIFY` command displays online change status for this IMS, which is participating in a local or global online change:

- Online change libraries
- Work in progress for resources to be deleted or changed by a `/MODIFY COMMIT` or a `INITIATE OLC PHASE(PREPARE)` command
- Resources to be added, changed, and deleted by a `/MODIFY COMMIT` or an `INITIATE OLC PHASE(COMMIT)` command

The status displayed for each library is displayed as `A` for active or `I` for inactive, and is followed by the DD names and databases that make up the library. Status information displays for the following libraries:

- IMSACBA and IMSACBB
- FORMATA and FORMATB
- MODBLKSA and MODBLKSB
- MATRIXA and MATRIXB

Work in progress for resources to be deleted or changed causes the `/MODIFY COMMIT` command to fail. You must wait for the work in progress to complete, or force it to complete (for example, by issuing a command), before issuing `/MODIFY COMMIT`. Table 49 on page 235 lists the resource type, resource name, and the status displays for work in progress that would prevent the `/MODIFY COMMIT`

command from completing successfully:

Table 49. Work in Progress For Resources that Prevent a Sucessful /MODIFY COMMIT

Resource Type	Resource Name	Status
DATABASE	dbname	AREAS OPEN:
		/DBD ACTIVE
		/DBR ACTIVE
		/DBR OR /STOP IN PROGRESS
		DATABASE IN USE
		DBR NEEDED FOR PARTITION
		nnnn PSB SCHEDULED
		RANDOMIZER: rndmname
		RECOVERY IN PROGRESS
AREA	areaname	OPEN
		RECOVERY IN PROGRESS
PROGRAM	pgmname	SCHEDULED
		QUEUING nn
		GLOBAL QUEUE COUNT INTERNAL ERROR
		GLOBAL QUEUE COUNT STORAGE ERROR
RTCODE	rtcode	ACTIVE
TRAN	tranname	IN USE
		QUEUING nn
		SCHEDULED
		SUSPENDED
		CONVERSATION TERM/USER
		nodename ID= nnnn
		nodename username ID= nnnn
		username ID= nnnn
		lin#-pte# ID= nnnn
		1- SC ID= nnnn
		GLOBAL QUEUE COUNT INTERNAL ERROR
		GLOBAL QUEUE COUNT STORAGE ERROR

In a shared-queues environment, `/DISPLAY MODIFY` only shows local work in progress, except for the TRAN QUEUING status and the PROGRAM QUEUING status, for which this command shows the sum of local and global queue counts. That is, `/DISPLAY MODIFY` does not show whether there is work in progress for other IMS subsystems sharing the message queues. If the `/DISPLAY MODIFY` command shows no work in progress for the IMS subsystem which the command was issued, a `/MODIFY COMMIT` command can be issued for that IMS subsystem.

If IMS cannot access the global queue counts, IMS displays the GLOBAL QUEUE COUNT INTERNAL ERROR status. If IMS cannot obtain sufficient storage to query global queue counts, IMS displays the GLOBAL QUEUE COUNT STORAGE ERROR status. In either case, online change will fail if there are global queue counts for resources being changed or deleted.

Table 50 lists the parameters that can be used with the `/DISPLAY MODIFY` command in a DBCTL environment.

Table 50. Parameters Supported in a DBCTL Environment

ADDS	DBS	MODS
ALL	DELS	PDS
CHNGS	DMS	PSS

Table 51 lists the parameters that can be used with the /DISPLAY MODIFY command in a DCCTL environment.

Table 51. Parameters Supported in a DCCTL Environment

ADDS	FMS	RCS
ALL	MODS	TRS
CHNGS	PDS	
DELS	PSS	

Examples of using the /DISPLAY MODIFY command are shown in “Example 1 for /DISPLAY MODIFY Command” on page 307.

/DISPLAY MODIFY with one or more parameters (not including ALL) is only valid if IMS is in the MODIFY PREPARE state, which means that a /MODIFY PREPARE or an INITIATE OLC PHASE(PREPARE) command completed successfully. The /DISPLAY MODIFY parameters and the information displayed are:

<blanks>
Displays status of online change libraries.

ADDS Displays the resources to be added by online change. Each display line contains a resource type, resource name, and a status of ADDED. The resource type can be: DATABASE, DMB, FORMAT, PROGRAM, PSB, RT CODE, or TRAN.

ALL Displays the status of online change libraries and any work in progress for database, program, routing code, or transaction resources to be deleted or changed by the /MODIFY COMMIT or the INITIATE OLC PHASE(COMMIT) command.

For DBCTL, transactions do not apply. For DCCTL, databases do not apply.

CHNGS
Displays the resources to be changed by online change. Each display line contains a resource type, resource name, and a status of CHANGED. Resource type can be: DATABASE, DMB, FORMAT, PROGRAM, PSB, RT CODE, or TRAN.

DBS Displays the databases to be changed or deleted with work in progress that would prevent /MODIFY COMMIT from succeeding. If DBS is specified with the ADDS, CHNGS, DELS, or MODS parameters, the databases in MODBLKS to be added, changed, or deleted display. Each display line contains a resource type of DATABASE, the database name, and a status of ADDED, CHANGED, or DELETED.

DELS Displays the resources to be deleted by online change. Each display line contains a resource type, resource name, and a status of DELETED. The resource type can be: DATABASE, DMB, FORMAT, PROGRAM, PSB, RT CODE, or TRAN.

DMS

If DMS is specified with the ADDS, CHNGS, DELS, or MODS parameters, the DMBs in ACBLIB to be added, changed, or deleted display as ADDED, CHANGED, or DELETED.

If the status of the associated database in MODBLKS is ADDED, DMS displays a DMB as ADDED. If the status of the associated database in MODBLKS is DELETED, DMS displays a DMB as DELETED.

FMS If FMS is specified with the ADDS, CHNGS, DELS, or MODS parameters,

the MFS formats to be added, changed, or deleted in FMTLIB display. Each display line contains a resource type of FORMAT, the format name, and a status of ADDED, CHANGED, or DELETED.

MODS Displays the resource to be modified by online change, which are resources to be added, changed, and deleted. Each display line contains the resource type, resource name, and a status of ADDED, CHANGED, or DELETED. The resource type can be: DATABASE, DMB, FORMAT, PROGRAM, PSB, RT CODE, or TRAN.

`/DISPLAY MODIFY MODS` shows all of the resources to be added, changed, or deleted by `/MODIFY COMMIT`, which is equivalent to the output from `/MODIFY PREPARE` with the LTERM keyword specified.

PDS Displays the programs to be changed or deleted in MODBLKS with work in progress that would prevent a `/MODIFY COMMIT` or an `INITIATE OLC PHASE(COMMIT)` command from succeeding. If PDS is specified with the ADDS, CHNGS, DELS, or MODS parameters, the programs in MODBLKS to be added, changed, or deleted display. Each display line contains a resource type of PROGRAM, the program name, and a status of ADDED, CHANGED, or DELETED.

PSS

If PSS is specified with the ADDS, CHNGS, DELS, or MODS parameters, the PSBs in ACBLIB to be added, changed, or deleted display as ADDED, CHANGED, or DELETED.

If the status of the associated program in MODBLKS is ADDED, PSS displays a PSB as ADDED. If the status of the associated program in MODBLKS is DELETED, PSS displays a PSB as DELETED.

RCS Displays the routing codes to be changed or deleted in MODBLKS with work in progress that would prevent a `/MODIFY COMMIT` or an `INITIATE OLC PHASE(COMMIT)` command from succeeding. If RCS is specified with the ADDS, CHNGS, DELS, or MODS parameters, RTCODEs in MODBLKS to be added, changed, or deleted display.

TRS Displays the transactions to be changed or deleted in MODBLKS with work in progress that would prevent a `/MODIFY COMMIT` or an `INITIATE OLC PHASE(COMMIT)` command from succeeding. If TRS is specified with the ADDS, CHNGS, DELS, or MODs parameters, transactions in MODBLKS to be added, changed, or deleted display.

`/DISPLAY MODIFY` displays these status terms:

AREAS OPEN:
A DEDB database has open areas. One or more lines listing the open areas follow this line.

/DBD ACTIVE
A `/DBDUMP` command is in progress for a database to be changed or deleted.

/DBR ACTIVE
A `/DBRECOVERY` command is in progress for a database to be changed or deleted.

/DBR OR /STOP IN PROGRESS
A `/DBRECOVERY` command or `/STOP` command is in progress for a DEDB database.

DATABASE IN USE
The database is in use by one or more active regions referencing the database.

DBR NEEDED FOR PARTITION
A /DBRECOVERY command must be issued for the DB partition before the database can be changed or deleted by online change

PSB SCHEDULED
A program referencing a database to be changed or deleted is scheduled.

Q STRUCTURE IS UNAVAILABLE
The queue structure is unavailable. IMS is unable to check the global queue count for transactions to be added, changed, or deleted. Commit is not permitted to proceed. The structure may be unavailable for one of the following reasons:

- CQS is unavailable
- The MSGQ structure failed and it hasn't been rebuilt yet
- CQS lost connectivity to the MSGQ structure

SCHEDULED
The named resource (a transaction or program to be changed or deleted, or a program referencing a database to be changed or deleted) is scheduled.

RECOVERY IN PROGRESS
A /RECOVER START command is in progress to recover one or more databases with the database recovery service.

RANDOMIZER: rndmname
A DEDB database that has no work in progress lists the randomizer name (rndmname). The randomizer display lines are information only, not work in progress that would prevent online change commit from succeeding.

This line is followed by a line with the randomizer name and status of LOADED or NOT LOADED.

If the randomizer is shared by one or more DEDBs, another line is displayed with 'DEDBS SHARING RANDOMIZER:' followed by the randomizer name. Online change won't delete a randomizer that is shared by other DEDBs. This line is followed by one line for each DEDB sharing the same randomizer.

ACTIVE
The routing code rtcode is active.

IN USE
Queuing is in progress for this transaction (either terminal input or program-to-program switch).

QUEUING nn
Messages are queued to the transaction or program; nn is the number of messages queued. Global queue counts represent the number of messages on the coupling facility list structure for the transaction or program.

SUSPENDED
The transaction to be changed or deleted is on the suspend queue.

CONVERSATION TERM/USER
The transaction to be changed or deleted or the transaction referencing a program or database to be changed or deleted is in conversation. The terminal originating the conversational transaction and the conversation ID are also displayed.

The terminal information displayed is:

- nodename for VTAM terminals
- nodename and username if an ETO user is signed on to the node
- username for ETO users that signed off while in conversation
- lin#-pte# (line and pterm number) for BTAM terminals
- 1-SC for the system console

/DISPLAY MSNAME

`/DISPLAY MSNAME` displays the queue counts for the specified logical link path. Generic parameters are supported for the MSNAME keyword.

An example of using the `/DISPLAY MSNAME` command is shown in "/DISPLAY MSNAME."

QCNT

Specifies that global queue count information is to be displayed. If `QCNT` is not specified, local queue counts are displayed; when it is specified, no local queue counts are displayed.

This keyword is only valid in a shared-queues environment.

/DISPLAY NODE

`/DISPLAY NODE`, when followed by a valid node name or ALL, displays status, queue counts, and number of messages sent to and received from the specified node. When using ISC, the output for a parallel-session node shows one line of information for each session of the node whether the node is active or not and whether logical terminals are assigned or not. If a session is active, the user identifier (preceded by a dash) and the status and queue counts are displayed on a subsequent line. `/DISPLAY NODE` displays queue counts up to a maximum of 32 KB. The queue count for `/DISPLAY NODE` is reset to zero when sessions are not active, or when a signed-on user issues `/SIGN OFF`.

The USER keyword is valid for ISC nodes with users or dynamic nodes with signed on users. The NODE parameter can be generic where the USER keyword is not present. The generic parameter specifies nodes that already exist. A status of STATIC is displayed for nodes defined to IMS via system definition. STATIC is also supported as an attribute parameter. Static nodes are used with the `/DISPLAY NODE` command in the same way that dynamic nodes are used.

`/DISPLAY NODE` shows signed on users, where the user ID follows the status SIGN in parentheses.

Examples of using the `/DISPLAY NODE` command are shown in

- "Example 1 for /DISPLAY NODE Command" on page 311
- "Example 2 for /DISPLAY NODE Command" on page 312
- "Example 3 for /DISPLAY NODE Command" on page 313
- "Example 4 for /DISPLAY NODE Command" on page 314
- "Example 5 for /DISPLAY NODE Command" on page 315
- "Example 6 for /DISPLAY NODE Command" on page 315
- "Example 7 for /DISPLAY NODE RECOVERY Command" on page 315

MODE The MODE keyword allows you to display the mode table names associated with the nodes specified. The output includes:

NODE-USR
node name and user identifier

TYPE type of node

DEF MODETBL
default mode table name set via system definition or logon descriptor or /CHANGE command. This name can be overridden using the /OPNDST command or the LOGON exit for non-IMS session initiations.

ACT MODETBL
mode table name actually used to initiate the session. This name is only displayed while the session is active. The field is blank at normal session termination.

EMHQ
Specifies that global queue count information on the Expedited Message Handler Queues (EMHQ) displays when both EMHQ and QCNT are specified. If EMHQ is specified, QCNT must also be specified. This keyword is only valid in a shared-queues environment.

QCNT
Specifies that global queue count information displays. If QCNT is not specified, local queue counts are displayed; when it is specified, no local queue counts are displayed. If QCNT is specified and EMHQ is not, global queue count information on the shared message queues (MSGQ) displays.

IMS does not check if the specified name is a valid LTERM in the IMS subsystem on which the command is entered.

This command does not display queue counts for Fast Path output messages for the NODE unless EMHQ and QCNT are specified.

This keyword is only valid in a shared-queues environment.

RECOVERY
The /DISPLAY NODE|USER RECOVERY command can be used to display the recovery values that pertain to the node or user. The recovery values may be displayed from the local control blocks, if available, or from values saved for the node or user in the Resource Manager, if resource information is being kept in Resource Manager, and the node or user is defined to have its status kept in Resource Manager. The recovery information that will be displayed with this command are:

- IMS owner

  The IMS ID (RSEname if an XRF system) of the IMS system that currently owns this resource. If the resource is not currently owned, this field will be displayed as NONE.
- Status Recovery Mode (SRM)

  The scope of recovery for a resource, and from where the end-user significant status is maintained and recovered.
- End-user significant status

  The following status are end-user significant status for nodes and users.
  - Conversation
  - STSN
  - Fast Path

These are statuses that frequently change for a resource, and thus there are performance considerations related to maintaining end-user significant status. The installation can tell IMS how to recover end-user significant status by specifying the level of recovery for each status. Shown for each resource is the level of recovery for each end-user significant status.

N/A is displayed by `/DISPLAY NODE ALL MODE` for VTAM 3270 nodes.

"Attribute" is a reserved parameter. Attributes listed in Table 46 on page 220 can be used with the NODE keyword. Nodes are selected for display based on the attribute or attributes specified. The attribute display format is the same as the standard NODE display. Attributes usually correspond to the conditions displayed for nodes. Any exceptions are flagged in the following list.

The attributes are reserved parameters for the `/DISPLAY NODE` command and cannot be used to name nodes.

The attributes that can be specified with the NODE keyword are listed in below. For example, `/DISPLAY NODE TRACE` displays all nodes that are currently being traced. Following is a list of the attributes that can be specified with the NODE keyword:

- ACTIV
- AUTOSR (Corresponds to the NODE conditions ASR)
- BCKUP
- CLSDST
- CON
- CONVACT (Corresponds to the NODE condition CONV-ACT)
- CONVHLD (Corresponds to the NODE condition CONV-HLD)
- C1INOP
- C2INOP
- C3INOP
- C4INOP
- DEACT
- EXCL
- FORCES (Corresponds to the NODE condition FORCE)
- IDLE
- INOP
- LOCK
- LOST
- MFST
- OPNDST
- PAGE
- PRI
- PRST
- QUI
- RECOVER
- RELREQ
- RESP
- RESPINP
- RESYNC

- SEC
- SHUT
- SIGN
- SIMLOGON
- STATIC
- STOPPED
- TEST
- TKOTRA
- TRA

Explanations of the attributes can be found in Appendix G, “Status and Attributes for the /DISPLAY Command,” on page 855.

/DISPLAY OASN SUBSYS

The /DISPLAY OASN SUBSYS command displays the outstanding recovery units (Origin Application Schedule Numbers) associated with the external subsystem (not CCTL subsystem) specified by the SUBSYS keyword.

The OASN is assigned by IMS when it schedules an application into a dependent region. The OASN, coupled with the IMS ID, becomes the recovery token for units of work distributed to other subsystems.

An example of using the /DISPLAY OASN SUBSYS command is shown in “Example for /DISPLAY OASN Command” on page 316.

/DISPLAY OLDS

/DISPLAY OLDS displays the system logging status. The ddnames for the online log data sets that are currently online are listed in the sequence used. That is, the one currently being used is listed first, with the one most recently used listed next, and so on. OLDS status information is displayed with each ddname. The meanings of these status messages are given below. In the case of dual logging, two lines of status output are displayed per OLDS ID; the first for the primary OLDS and the second for the secondary OLDS.

/DISPLAY OLDS shows the percentage of the OLDS used (shown under % FULL in the output) and the rate at which it is being filled (shown under RATE in the output). The RATE is the average number of blocks written to the current OLDS since the last /DISPLAY OLDS was issued. If no blocks were written, the RATE is 0, or, if elapsed time is 0, the previous RATE will be displayed.

An example of using the /DISPLAY OLDS command is shown in “Example for /DISPLAY OLDS Command” on page 316.

/DISPLAY OLDS shows one of the following as the archive status of each OLDS:

NEEDED
This OLDS needs to be archived. It cannot be reused for output logging until it has been archived.

SCHEDULED
An archive job has been generated for this OLDS.

STARTED
A job to archive this OLDS has started execution.

AVAILABLE
Either this OLDS is empty or it has been archived. It can be reused for output logging.

When the archive status of an OLDS is scheduled or started and DBRC has generated the JCL for the archive job, the archive job name will be included in the output.

`/DISPLAY OLDS` also shows the following status information for OLDS:

IN USE This is the current output log OLDS. The display line for this OLDS also has an asterisk (*) in the left margin.

STOPPED `/STOP` command entered for this OLDS or the OLDS is being internally stopped because of write errors.

WRT-ERR A write I/O error occurred on this OLDS data set.

CLSE An error occurred when closing this data set. The Log Recovery utility can be used to correct this condition.

PREV CLSER The previous OLDS could not be closed because of an I/O error and this OLDS is required by the Log Recovery utility to correct the condition.

NOT USABLE This data set is not being used for logging. When operating in "Degraded Dual" state, the other data set in this OLDS pair was used for output but this data set was not used because of a previous data set error.

BACKOUT This OLDS is potentially required for backout.

When the backout of at least one dependent region would require an SLDS because all of its log records are not available from OLDS, the following line will be inserted in front of `SLDSREAD ON` or `SLDSREAD OFF` in the OLDS STATUS display:

```
SLDS REQUIRED FOR BACKOUT - RGN nnnnn
```

The value nnnnn is the region number of the application that would require the oldest record for backout. An OLDS that is stopped or had an I/O error will be dynamically deallocated after it is no longer needed for backout.

Following the detailed OLDS data set information, the dual OLDS state (DUAL, DEGRADED DUAL, NONDEGRADABLE DUAL or SINGLE) and the WADS recording state (DUAL, SINGLE, or NO) display. Also, the DDNAMEs of all available WADS display. The DDNAMEs of the current WADS are preceded by an asterisk (*).

SLDSREAD ON or OFF is also displayed. If SLDSREAD is on it indicates that IMS is enabled to retrieve both SLDS and OLDS.

Restriction: Under the dual logging environment, the number of primary OLDS and secondary OLDS must be the same; otherwise, the status of the OLDS is incorrect.

The following parameters, when used with the /DISPLAY OLDS command, specify selection criteria used to select which OLDS are to be included in the display output. All these parameters are optional. If none are specified, ALL is the default action. When multiple parameters are specified, an OLDS satisfying any one of the specified criteria is included in the output.

ALL Display status of each online OLDS.

INUSE
Display current output OLDS.

BACKOUT
Display OLDS containing dynamic backout data for currently active PSTs.

ERROR
Display OLDS that have encountered error conditions.

STOPPED
Display stopped OLDS.

/DISPLAY OTMA

/DISPLAY OTMA displays the current status for IMS Open Transaction Manager Access (OTMA) clients and servers. This command displays the following:

- Each member in each XCF group

  The server is always the first member displayed.
- The XCF status for each member

  The status can be one of the following:
  - ACTIVE
  - CREATED
  - FAILED
  - NOT DEFINED
  - QUIESCED
  - UNKNOWN

  If the server leaves the XCF group (for example, because of a /STOP OTMA command), then the XCF status is displayed as NOT DEFINED.
- The user status for each member

  The status can be one of the following:
  - ACCEPT TRAFFIC
  - DISCONNECTED
  - IN SLOWDOWN
  - SERVER
  - WAIT BID
  - WAIT RESPONSE
- The security status for each server

  The security status can be one of the following:

  ```
  NONE
  CHECK
  FULL
  PROFILE
  ```

  See Chapter 53, “/SECURE,” on page 583 for more information on the security status.

This command is functionally equivalent to the /DISPLAY TMEMBER ALL command.

Examples of using the /DISPLAY OTMA command are shown in:

- "Example 1 for /DISPLAY OTMA Command" on page 316
- "Example 2 for /DISPLAY OTMA Command" on page 316

/DISPLAY OVERFLOWQ

/DISPLAY OVERFLOWQ displays a list of queue names that are in overflow mode for those coupling facility list structures specified (for the STRUCTURE keyword) that are in overflow mode. It also displays the queue type of the resource for each queue. If a specified structure is not defined, the overflow structure name is displayed as NOT-DEFINED.

Queue names displayed might not be defined to the IMS subsystem on which this command is issued. If a queue type is not defined, the type is displayed as UNKNOWN, and the queue name is displayed as the 16-byte queue name (with the queue type as the first byte).

This command is only valid in a shared-queues environment.

STRUCTURE
Specifies the primary coupling facility list structures to be displayed.

This command displays the following information:

APPC The resource type on the overflow structure is an APPC outbound queue. The first eight bytes of the resource name displayed is the unique constant TMSTMP-. The next four bytes are the IMS ID of the IMS subsystem that has the LUNAME and TPNAME defined. If you issue the /DISPLAY OVERFLOWQ command on the system on which the LUNAME and TPNAME are defined, then the LUNAME and TPNAME also display.

When LUNAME and TPNAME cannot be located on the system on which they're defined, UNKNOWN appears under the LUNAME and TPNAME headings in the display output.

BALGRP The resource type on the overflow structure is a Fast Path PSB name. The resource name displayed is the Fast Path PSB name.

LTERM The resource type on the overflow structure is an LTERM. The resource name displayed is the LTERM name.

OTMA The resource type on the overflow structure is an OTMA outbound queue. The first eight bytes of the resource name displayed is the unique constant TMSTMP-. The next four bytes are the IMS ID of the IMS subsystem that has the TMEMBER and TPIPE defined. If you issue the /DISPLAY OVERFLOWQ command on the system on which the TMEMBER and TPIPE are defined, then the TMEMBER and TPIPE are also displayed.

When LUNAME and TPNAME cannot be located on the system on which they're defined, UNKNOWN appears under the LUNAME and TPNAME headings in the display output.

REMOTE The resource type on the overflow structure is a remote LTERM,

remote transaction, or an MSNAME. The resource name displayed is the remote LTERM name, remote transaction name, or the msname.

SUSPENDTRAN
The resource type on the overflow queue is a suspended transaction and the messages are on the suspend queue. The resource name displayed is the transaction name. If the transaction that is suspended is a serial transaction, or if local affinity has been set for a non-serial transaction, the first eight bytes displayed are the transaction name and the last seven bytes represent the IMS ID of the subsystem that enqueued the message.

SERIALTRAN The resource type on the overflow queue is a serial transaction. The first eight bytes of the resource name are the transaction name and the last seven bytes represent the IMS ID of the IMS subsystem that inserted the message for the transaction.

TRANSACTION
The resource type on the overflow queue is a transaction. The resource name displayed is the transaction name. If local affinity has been set for the transaction, the first eight bytes displayed are the transaction name and the last seven bytes represent the IMS ID of the IMS subsystem that enqueued the message.

UNKNOWN The resource type of the resource on the overflow queue is not known to this IMS subsystem. The 16 byte resource name is displayed.

An example of using the /DISPLAY OVERFLOWQ command is shown in “Example for /DISPLAY OVERFLOWQ Command” on page 316.

/DISPLAY POOL

/DISPLAY POOL displays processor storage utilization statistics for the various IMS storage pools.

When the /DISPLAY POOL command is issued from a specific environment (such as DBCTL), only the information that is valid for that environment is displayed.

Certain storage utilization statistics can be obtained with the DL/I STAT call.

Control block table (CBT) pools are defined in DFSCBT00 and are grouped into classes for the purposes of the /DISPLAY POOL command. All CBT pools are valid in all environments.

Examples of using the /DISPLAY POOL command are shown in:

- “Example 1 for /DISPLAY POOL Command” on page 317
- “Example 2 for /DISPLAY POOL Command” on page 320
- “Example 3 for /DISPLAY POOL Command” on page 320
- “Example 4 for /DISPLAY POOL Command” on page 321
- “Example 5 for /DISPLAY POOL Command” on page 322
- “Example 6 for /DISPLAY POOL Command” on page 323
- “Example 7 for /DISPLAY POOL Command” on page 323
- “Example 8 for /DISPLAY POOL Command” on page 323
- “Example 9 for /DISPLAY POOL Command” on page 324

- "Example 10 for /DISPLAY POOL Command" on page 324
- "Example 11 for /DISPLAY POOL Command" on page 324
- "Example 12 for /DISPLAY POOL Command" on page 324
- "Example 13 for /DISPLAY POOL Command" on page 324
- "Example 14 for /DISPLAY POOL Command" on page 325
- "Example 15 for /DISPLAY POOL Command" on page 325

The classes and corresponding parameters are:

CBT This displays storage for all CBT pools. Use of this operand generates a large amount of display output.

DBB This displays database related storage.

DCC This displays storage associated with DC.

DEP This displays storage associated with the dependent region structure. This parameter is not supported for an RSR tracking subsystem.

DISP This displays dispatcher related storage.

FP This displays Fast Path related storage.

GEN This displays storage that has no particular functional association.

OSAM This displays OSAM related storage.

SUM This displays summary statistics for each of the above storage classes.

A CBT class display shows information for each pool in that class. The amount of storage displayed for these pools might change because the storage is obtained as needed and can be freed when not in use. In addition, a summary of the total current global and local storage in the class and in all CBT pools is displayed.

If SUM is specified, a summary of the total current local and global storage for each CBT class is displayed.

For the following storage pools, the amount of storage obtained is specified during system definition or via EXEC statement overrides in the IMS procedure JCL.

DBAS Database buffer pools (includes VSAM buffer pools if VSAM is in the system, also includes buffer space used by sequential buffering).

Statistics for VSAM local shared resource pools are displayed in the order in which the pools are defined. For each local shared resource pool, each subpool's statistics are displayed in ascending order by buffer size. The smallest buffers are listed first and the largest buffers are listed last. If an index subpool exists in the shared resource pool, its statistics follow the data subpool statistics. Index statistics are also displayed in ascending order by buffer size.

Headings for VSAM subpools include the pool ID of the local shared resource pool. They also indicate whether the subpool is a data subpool (type D) or an index subpool (type I).

DBWP

Database working pool.

DMBP Data management block pool.

EPCB Fast Path PCB extension pool for MPPs.

FPDB Fast Path database buffer pool.

MAIN Working storage pool.

MFP Message format block pool.

PSBP Program specification block pool This parameter is not supported for an RSR tracking subsystem.

PSBW Program specification block work area. This parameter is not supported for an RSR tracking subsystem.

QBUF Message queue buffer pool. Parameter QBUF can also be used to indicate how close the system is to automatic shutdown due to message queue buildup. The statistics displayed for each data set are:

- The maximum number of records available before initiation of automatic shutdown. This value is defined by the SHUTDWN parameter of the MSGQUEUE macro. This value is not the total number of records available in the data set.
- The number of records currently in use.
- The percentage of records currently in use to records available before initiation of automatic shutdown. When the value described in this item reaches 100%, an internal checkpoint dumpq will automatically be initiated. The number of records in the queue data set might continue to increase during the checkpoint dumpq, resulting in percentages greater than 100 being displayed.

Table 52 shows the environments (DB/DC, DBCTL, or DCCTL) in which each storage pool is valid. If a pool is not valid in an environment, it will not appear when `/DISPLAY POOL ALL` is issued.

Table 52. Storage Pools and Their Environments

Storage Pools	DB/DC	DBCTL	DCCTL
AOIP	X	X	X
CESS	X	X	X
CIOP	X	X	X
DBAS	X	X	
DBWP	X	X	
DMBP	X	X	
EMHB	X	X	X
EPCB	X	X	X
FPDB	X	X	
FPWP	X	X	
HIOP	X		X
LUMC	X		X
LUMP	X		X
MAIN	X	X	X
MFP	X		X
PSBP	X	X	X
PSBW	X	X	X
QBUF	X		X

Table 52. Storage Pools and Their Environments (continued)

Storage Pools	DB/DC	DBCTL	DCCTL
Note:			
When /DISPLAY POOL ALL is specified for an RSR tracking subsystem, the following pools are not included: DEP, PSBP, PSBW, LUMP, LUMC.			

Abbreviations Used in the /DISPLAY POOL Command

The following lists indicate the abbreviations used in the /DISPLAY POOL command for each pool and subpool.

CBT Pool Display

NAME Storage pool name

SP Associated MVS/ESA user number

CURR Current bytes allocated to the pool

MAX Maximum size in bytes that the pool ever reached

GETS Number of GETMAINs issued

FREES Number of FREEMAINs issued

Message Queue Pool

BFRS Number of processor storage queue buffers

SIZE Usable size of one queue buffer, excluding the buffer prefix

ENQ Number of message enqueues

DEQ Number of message dequeues

CAN Number of canceled messages

WAIT Number of I/O waits issued

2 **I/O** Number of I/O operations. In a shared queues
2 environment, it is the sum of CQSPUTs and
2 CQSREADs.

ERR Number of I/O errors

Message Format Pool

SIZE Pool size

SPACE Maximum amount of free space

DIRS Space used by MFS Dynamic Directory

REQ1 Number of block requests from pool

I/O Number of fetch I/O operations

DIR Number of directory I/O operations

WAIT Number of immediate fetch I/O operations

FREE Amount of free space currently in the pool

ERR Number of I/O errors

Database Buffer Pools

The database buffer pool display consists of the OSAM buffer pool and, if VSAM is in the system, the VSAM buffer subpool. The display also includes information about the buffer space used by sequential buffering within the online IMS subsystem. The information about the sequential buffering buffer space includes:

MAX Maximum amount of space available for sequential buffering

FREE Amount of currently free space for sequential buffering

CUR Current bytes allocated to the pools for sequential buffering

HIGH Highest amount of space ever used in this run for sequential buffering

OSAM Buffer Subpool

Product-sensitive programming interface

Statistics for each OSAM subpool are displayed separately. The final display provides the statistical summation for all the OSAM subpools in all the shared resource pools.

SIZE Buffer pool size

REQ1 Number of block requests

REQ2 Number of requests satisfied in the pool plus new blocks created

READ Number of read requests issued

BISAM Number of BISAM reads issued plus QISAM SETLs

WRITES Number of OSAM writes issued

KEYC Number of retrieve by key calls number

LCYL Number of OSAM format logical cylinder requests

PURG Number of synchronization calls received

OWNRR Number of release ownership requests

ERRORS Number of permanent errors now in the pool and largest number of permanent errors during this run

End of Product-sensitive programming interface

VSAM Buffer Subpool

Product-sensitive programming interface

Statistics for each VSAM subpool are displayed separately. The final display provides the statistical summation for all the VSAM subpools in all the shared resource pools.

BSIZE The size of the buffers in this VSAM subpool

POOLID Local shared resource pool ID

TYPE	The subpool type: I (Index) or D (Data)
RRBA	Number of retrieval requests by RBA
RKEY	Number of retrieval requests by key
BFALT	Number of logical records altered
NREC	Number of new VSAM logical records created
SYNPTS	Number of system checkpoint requests
NMBUFS	Number of buffers in this VSAM subpool
VRDS	Number of VSAM control interval reads
FOUND	Number of control intervals VSAM found in the subpool through lookaside
VWTS	Number of VSAM control interval writes
ERRORS	Total number of permanent errors now in the pool and total number of permanent errors in this run

The BSIZE value in the final display is ALL.

End of Product-sensitive programming interface

Enhanced OSAM Buffer Subpool

Product-sensitive programming interface

Each OSAM subpool is displayed separately. The final display provides the statistical summation for all the OSAM subpools.

ID	The 4-character POOLID provided at subpool definition time
BSIZE	The size of the buffers in this OSAM subpool
NBUF	Number of buffers for this subpool
FX=	The fix options for this subpool Y/N indicates whether or not the DATA BUFFER PREFIX/DATA BUFFERS are fixed.
LCTREQ	Number of LOCATE-type calls for this subpool
ALTREQ	Number of buffer alter calls for this subpool. This count includes NEW BLOCK and BYTALT calls.
PURGRQ	Number of PURGE calls for this subpool
FNDIPL	Number of LOCATE-type calls, for this subpool, where data is already in the OSAM pool
BFSRCH	Number of buffers searched by all LOCATE-type calls for this subpool
RDREQ	Number of READ I/O requests for this subpool
BFSTLW	Number of single block writes initiated by buffer steal routine for this subpool
PURGWR	Number of buffer written by purge
WBSYID	Number of LOCATE calls, for this subpool, that waited due to busy ID

WBSYWR	Number of LOCATE-type calls, for this subpool, that waited due to busy writing
WBSYRD	Number of LOCATE-type calls, for this subpool, that waited due to busy buffer reading
WRLSEO	Number of buffer steal or purge requests, for this subpool, that waited for ownership to be released.
WNOBFR	Number of buffer steal requests, for this subpool, that waited because no buffers were available to be stolen
ERRORS	Total number of I/O errors for this subpool, or number of buffers locked in the pool due to write errors for this subpool
CF-READS	Number of blocks read from CF.
EXPCTD-NF	Number of blocks expected but not read.
CFWRT-PRI	Number of blocks written to CF (prime).
CFWRT-CHG	Number of blocks written to CF (changed).
STGGLS-FULL	Number of blocks not written (STG CLS full).
XI-CNTL	Number of XI buffer-invalidate calls.
VECTR-XI	Number of buffers found invalidated by XI on VECTOR call.
SB-SEQRD	Number of immediate (SYNC) sequential reads (SB stat).
SB-ANTICIP	Number of anticipatory reads (SB stat).

For the summary totals, the POOLID is not shown. BSIZE is set to ALL. NBUF indicates the total number of buffers in the pool. The FX= field is replaced by OSM=, which shows the total size of the OSAM buffer pool.

End of Product-sensitive programming interface

Enhanced VSAM Buffer Subpool

Product-sensitive programming interface

Each VSAM subpool is displayed separately. The final display provides the statistical summation for all the VSAM subpools.

ID	The 4-character POOLID provided at subpool definition time
BSIZE	The size of the buffers in this VSAM subpool
TYPE	identifies the subpool as containing INDEX or DATA buffer
FX=	Number of options for this subpool. Y/N is used to indicate whether or not the INDEX BUFFERS/DATA BUFFER PREFIX/ DATA BUFFERS are fixed.
RRBA	Number of retrieval requests by RBA
RKEY	Number of retrieval requests by KEY

BFALT	Number of logical records altered
NREC	Number of new VSAM logical records created
SYNC PT	Number of system checkpoint (synchronization point) requests
NBUFS	Number of buffers in this VSAM subpool
VRDS	Number of VSAM control interval reads
FOUND	Number of control intervals VSAM found in the subpool through lookaside
VWTS	Number of VSAM control interval writes
HSR-S	Number of successful VSAM reads from Hiperspace buffers
HSW-S	Number of successful VSAM writes to Hiperspace buffers
HS NBUFS	Number of Hiperspace buffers defined for this subpool
HS R/W-FAIL	Number of failed VSAM reads/writes from or to Hiperspace buffers. This indicates the number of times a VSAM READ/WRITE request from or to Hiperspace resulted in DASD I/O.
ERRORS	Number of permanent write errors now in the subpool, or the largest number of errors in this execution

The BSIZE value in the final display is the total size of all the VSAM subpool buffers. BSIZE is set to ALL. For the summary totals, the POOLID is not shown. The TYPE and FX= fields are replaced by VS=, which is the total size of the VSAM subpool in virtual storage, and HS=, which is the total size of the VSAM subpool in Hiperspace.

End of Product-sensitive programming interface

Fast Path Database Buffer Pool

AVAIL	Number of available page-fixed database buffers
WRITING	Number of page-fixed database buffers being written to disk, including the buffers being filled with sequential dependent segments
PGMUSE	Number of page-fixed buffers allocated to PSTs
UNFIXED	Number of available buffers for starting a new region
POOLNAME	Name of the private buffer pool for the area. This name is specified in the VSPEC member.
CISIZE	Control interval size of the private pool for the area.
PBUF	Size allocated for the primary pool (specified in the VSPEC member).
SBUF	Size allocated for the secondary pool (specified in the VSPEC member).

MAX Maximum number of buffers for the private pool (specified in the VSPEC member).

CURRENT Total number of buffers currently in the pool, including both primary and secondary allocations.

LK `Y` indicates that buffer lookaside is active for the pool; `N` indicates that it is not.

HITS Percentage of searches of the pool for which a buffer was found. Only displayed for lookaside pools.

VALID Percentage of times a buffer found in the pool had valid data. An Invalid buffer is read from DASD and the pool copy is replaced. Only displayed for lookaside pools.

You use the percentages displayed for HITS and VALID together.

Example: If HITS is 40%, and VALID is 75%, a buffer was found in the pool 40% of the time, and of that 40%, 75% of the buffers found had valid data, that is, 30% of the HITS had valid data. So, IMS had to read data from DASD approximately 70% of the time.

PSBP, DMBP, PSBW, DBWP, EPCB, and MAIN Buffer Pools

SIZE pool size

FREE amount of currently free space

HIGH Highest amount of space ever used in this run

AOIP, CIOP, HIOP, CESS, FPWP, EMHB, LUMP, and LUMC Buffer Pools

During the execution of IMS, the AOIP, CIOP, HIOP, CESS, FPWP, EMHB, LUMP, and LUMC buffer pools can dynamically expand and contract.

Internally, each dynamic pool has one or more blocks of storage that are not contiguous, each divided into fixed length buffers. By obtaining new blocks and releasing unused blocks, a buffer pool can expand or contract during the execution of IMS. The number of blocks needed for each pool depends on several factors such as the number of buffer requests from the pool, buffer size, and block size.

SIZE Pool size

HIGH Maximum size since last checkpoint

LIMIT Upper expansion limit

OVERFLOW Size of the oversized chain

/DISPLAY PROGRAM

`/DISPLAY PROGRAM` displays the status of programs.

The status displayed can be one of the following:

```
DB-STOPD
I/O PREVEN
LOCK
NOTINIT
STOPPED
TRA
```

For definitions of the status conditions, refer to Appendix G, “Status and Attributes for the /DISPLAY Command,” on page 855.

For DBCTL, this command displays the status of BMP programs, DEDB utilities, and CCTL PSBs. Because DBCTL has no knowledge of CICS programs, their status is not displayed.

An example of using the /DISPLAY PROGRAM command is shown in “Example for /DISPLAY PROGRAM Command” on page 325.

/DISPLAY PSB

/DISPLAY PSB displays the status of PSBs, which transactions these PSBs are processing, any routing code associated with the transactions, the databases being accessed, and the type of access. This keyword can be used only if Fast Path is installed.

For HSSP PSBs, the letter H is appended to the access intent under the ACCESS heading.

For DBCTL, the status of CCTL PSBs is displayed.

An example of using the /DISPLAY PSB command is shown in “Example for /DISPLAY PSB Command” on page 326.

/DISPLAY PTERM

/DISPLAY PTERM can be specified without an associated LINE keyword on /DISPLAY when PTERM is used with the special “attribute” parameter. Physical terminals are selected for display based on the attribute or attributes specified. The attribute display format is the same as the standard LINE/PTERM display. Attributes usually correspond to the conditions displayed for line/physical terminals.

An example of using the /DISPLAY PTERM command is shown in “Example for /DISPLAY PTERM Command” on page 327.

The attributes that can be specified with the PTERM keyword are listed in Table 53. For example, /DISPLAY PTERM TRACE displays all physical terminals that are currently being traced.

Table 53. /DISPLAY PTERM Command Attributes

COMPINOP	CONVACT[1]
CONVHLD[2]	EXCL
INOP	LOCK
LOOPTEST	MFSTEST
NOIN	NOOUT
NOQUEUE	NOTOPEN
PAGE	PRST
PSTOPPED	PUR

Table 53. /DISPLAY PTERM Command Attributes (continued)

RESP	RESPINP
SIGN	STOPPED
TEST	TKOTRA
TRA	

Notes:

1. Corresponds to the NODE condition CONV-ACT.
2. Corresponds to the NODE condition CONV-HLD.

/DISPLAY Q

`/DISPLAY Q` displays the message queues according to classes and priority levels.

BALGRP
Displays queue counts of the Fast Path load balancing group queues.

CLASS
Displays queues for specified classes.

PRIORITY
Displays queues for specified priorities.

QCNT
Specifies that global queue count information is to be displayed for Fast Path load balancing groups. If `QCNT` is not specified, only local queue counts are displayed; when it is specified, no local queue counts are displayed.

This keyword is only valid in a shared-queues environment.

TRANSACTION
Displays queues for transactions that are ready to run. DFSCPIC is displayed as psbname for CPI Communications driven transactions that have not issued a DL/I call to allocate a PSB.

Combinations of the CLASS, PRIORITY, and TRANSACTION keywords allow classes, priority levels within classes, transactions within priority levels, and message counts to be displayed on an as-needed basis.

Examples of using the `/DISPLAY Q` command are shown in:

- "Example 1 for /DISPLAY Q Command" on page 327
- "Example 2 for /DISPLAY Q Command" on page 329

/DISPLAY QCNT

`/DISPLAY QCNT` displays global queue information for the specified resource type. The resource type can be APPC, BALGRP, LTERM, OTMA, REMOTE, or TRANSACTION. This command displays all the queues for the resource type with at least one message whose message age is greater than the message age value specified.

The following list describes the resource types allowed:

APPC Specifies that global information for all APPC outbound queues is to be displayed.

For messages that are placed on the shared queues using a side information entry name, the side information entry name is returned in the LUNAME field and the character string, DFSSIDE, is returned in the TPNAME field.

BALGRP
Specifies that global information for LTERMs and MSNAMEs is to be displayed.

LTERM
Specifies that global information for LTERMs and MSNAMEs is to be displayed.

MSGAGE
Specifies the message age in number of days (0-365). Only those queues with messages older than or equal to the value are displayed. The resource name and the time when the message was placed on the shared queues are displayed.

If you specify MSGAGE 0, all resource (queues) are displayed.

OTMA Specifies that global information for all OTMA outbound queues is to be displayed.

REMOTE
Specifies that global information for remote transactions and remote LTERMs is to be displayed.

TRANSACTION
Specifies that global information for transactions is to be displayed, not including transactions that are suspended.

Recommendation: When you issue this command, IMS reads every message for the resource type. To minimize the performance impact, issue this command only when necessary.

This command is valid only in a shared-queues environment.

The output from the /DISPLAY QCNT command contains the following information:

QUEUENAME	1-8 byte queue name. If the output is the result of a /DISPLAY QCNT APPC MSGAGE or a /DISPLAY QCNT OTMA MSGAGE command, the value displayed for the queue name is the character string 'TMSTMP' followed by the IMSID (or the first seven bytes of the RSENAME if XRF capable) of the IMS to which the output messages have affinity.
QCNT-TOTAL	Total count of messages on the queue.
QCNT-AGED	Count of messages with a message age greater than or equal to the message age specified. This count does not include messages whose age is less than the message age specified.
TSTMP-OLD	The timestamp of the oldest message for the queuename on the Shared Queue.
TSTMP-NEW	The timestamp of the newest message for the queuename on the Shared Queue.

3 **Note:** The timestamp is the time when the message
3 is put on the Shared Queues by CQS. These
3 timestamps can be used to select records from the
3 CQS log or the IMS log. Because this is a CQS
3 generated timestamp, it may not correspond to the
3 timestamp for the same message on the IMS log as
3 logging a message in IMS and logging the same
3 message in CQS are separate events.

Examples of using the /DISPLAY QCNT command are shown in “Example for /DISPLAY QCNT Command” on page 329.

/DISPLAY RECOVERY

2 The /DISPLAY RECOVERY command displays the recovery process in progress. It also
2 displays the list of database data sets and areas being recovered by an online
2 database recovery service. The database data sets and areas are displayed as part
2 of a *recovery list*, which is a set of database data sets, areas, or both processed by
2 an online database recovery service in a single recovery instance. The /DISPLAY
2 RECOVERY command also shows the status of one or all of the recovery lists that
2 exist.

If an online database recovery service is executing in an IMS DB/DC control region, the /DISPLAY RECOVERY command can be issued from LTERMs, automated operator (AO) application programs, the IMS Master Terminal (MTO), or the MVS master console and secondary console.

If an online database recovery service is running in an IMS DBCTL region, the /DISPLAY RECOVERY command can be issued by programs using the IMS DBCTL AOI or the MVS master console and secondary console.

Examples of using the /DISPLAY RECOVERY command are shown in:

- “Example 1 for /DISPLAY RECOVERY Command” on page 330
- “Example 2 for /DISPLAY RECOVERY Command” on page 330
- “Example 3 for /DISPLAY RECOVERY Command” on page 331
- “Example 4 for /DISPLAY RECOVERY Command” on page 331
- “Example 5 for /DISPLAY RECOVERY Command” on page 332
- “Example 6 for /DISPLAY RECOVERY Command” on page 332
- “Example 7 for /DISPLAY RECOVERY Command” on page 332
- “Example 8 for /DISPLAY RECOVERY Command” on page 333

ALL
An optional parameter that displays all the recovery list information and the recovery progress information. No recovery list entry information is displayed.

RCVTOKEN
An optional parameter that displays the recovery token associated with the specific recovery list. The recovery list information and recovery progress information for the specific list display with the recovery list entry information, including the database data set and areas contained in the recovery list.

token
Specifies the unique recovery token associated with the recovery list. This token can be up to eight characters in length. If RCVTOKEN is specified, *token* must be supplied.

The /DISPLAY RECOVERY command output is divided into three sections.

- Recovery List Information
- Recovery Progress Information
- Recovery List Entry Information

Recovery List Information

This section displays the status of one or more recovery lists. The status includes the following information:

TOKEN
This shows the recovery list token. If /DIS RECOVERY ALL is issued and no recovery lists exist, NO LISTS is displayed. If IMS is unable to obtain recovery list information from the recovery facility, UNAVAIL is displayed.

STATUS
The current status of the recovery list. The status is one of the following:

FORMING
A /RECOVER START command has not been issued.

STARTED
A /RECOVER START command has been issued. Recovery processing started but has not completed.

UNKNOWN
No list with the given token can be found.

STOPPING
A /REC STOP ALLENT command was issued.

ERROR
The choices are ABORT or CONT. These are the action options in case an error is encountered.

REC TYPE
Indicates the type of recovery being processed. The type is one of the following:

FULL A full recovery is processing.

TSR A Timestamp Recovery is processing. This occurs when RCVTIME was specified on the /RECOVER START command, but PITR was not specified.

PITR A Timestamp Recovery is processing with the point-in-time recovery (PITR) option. This occurs when RCVTIME is specified on the /RECOVER START command with PITR.

N/A Recovery has not been started. The type is undefined.

PROC The values that may appear reflect the recovery progress in two phases of processing: reading log data sets and restoring image copy data. The format of the entries are as follows:

- nnnn of mmmm LOGS READ
- nnnn of mmmm RESTORED
- COMPLETE - this state will only appear for a few seconds following the completion of restoration of the last DBDS and termination of the recovery list.
- NOT STARTED - The recovery list is ready for processing or for additional updates.

2 • routing time - if an online database recovery service is being used.

2 **IC#** Indicates the primary image copy will be used if an image copy is to be
2 used during recovery.

2 **SOURCE**
2 The primary copy as contained in the RECON is used for recovery.

Recovery Progress Information

This section displays status of the recovery in progress, if there is one. The section is omitted if no recovery is in progress. The status includes the following information:

TOKEN
The recovery list token.

LAST PROCESSED
The timestamp of the last log record processed by the recovery. The time is displayed in local time.

RCVTIME
The RCVTIME value specified on the /RECOVER START command or N/A if none was specified. The time is displayed in local time.

Recovery List Entry Information

This section displays the list of database data sets and areas in the given recovery list. The status includes the following information:

DATABASE DATA SET
For full function, this is the DB name and DD name of the database data set. For Fast Path it is the area name. If there are multiple area data sets for the area, the string '(MADS)' will also be displayed.

START OPTION
3 This shows the start option that was specified on the /RECOVER ADD or
3 /RECOVER START command or taken from the system default. The values can
3 be **STALOCAL**, **STAGLOBAL**, or **OFFLINE**.

STATUS
This indicates the status of the database data set or area. The status can be one of the following:

FAILED
An error occurred for this database data set or area during recovery.

STOPPED
A /RECOVER STOP command was entered to stop recovery processing for the database data set or area.

NORMAL
Recovery is in progress or pending for the database data set or area.

INVALID
DBRC no longer contains information about this database data set or area.

AUTH SSID
A list of SSIDs that are still authorized to the database data set or area. **NONE** displays if no IMS is authorized to the database data set or area. If recovery is in progress, **N/A** displays.

/DISPLAY RTCODE

/DISPLAY RTCODE displays the status of Fast Path routing codes, the PSB using the routing code, and the region, if any.

An example of using the /DISPLAY RTCODE command is shown in "Example for /DISPLAY RTCODE Command" on page 333.

/DISPLAY SHUTDOWN STATUS

/DISPLAY SHUTDOWN STATUS displays system activity during a shutdown type of checkpoint. When issuing the /DISPLAY SHUTDOWN STATUS command from a specific environment (such as DBCTL), only the information that is valid for that environment is displayed. For example, in a DBCTL environment, only the number of IMS threads is displayed.

In an LU 6.2 environment, the /DISPLAY SHUTDOWN STATUS command indicates the dependent regions that have active CPI Communications driven transaction programs. These regions must be terminated prior to shutdown completion. In addition, the command also displays LU 6.2 conversations that hang the shutdown processing.

In a shared-queues environment, the /DISPLAY SHUTDOWN STATUS command displays the CQS job name that needs to be restarted if IMS shutdown hangs because CQS is inactive.

If a /DISPLAY SHUTDOWN STATUS command is issued while a database is being recovered with the Online Recovery Services, the response DB RECOVERY IN PROGRESS will be returned in addition to the other responses.

Examples of using the /DISPLAY SHUTDOWN STATUS command are shown in

- "Example 1 for /DISPLAY SHUTDOWN STATUS Command" on page 333
- "Example 2 for /DISPLAY SHUTDOWN STATUS Command" on page 334
- "Example 3 for /DISPLAY SHUTDOWN STATUS Command" on page 334

The status displayed when the /DISPLAY SHUTDOWN STATUS command is issued can include the following:

xx MSG PROCESSING REGIONS(S) ACTIVE
Where xx is the number of BMP, TP, and FP regions that are active. If the checkpoint has already been posted, this is the only status displayed.

For DBCTL, a count of active CCTL threads is included.

SYSTEM PURGING
The checkpoint is the result of a /CHECKPOINT PURGE command.

TERMINAL USER STATUS
The header for the portion of the display that shows active terminals, whose status can be one of the following:

```
INPUT IN PROCESS
OUTPUT IN PROCESS
AWAITING RESPONSE
XX MSGS IN QUEUE
```

If there is no terminal activity, the status can be one of the following:

```
NO INPUTTING LINES
NO OUTPUTTING LINES
```

MSG-IN X MSG-OUT Y

The message totals, where X is the total number of input messages and Y is the total number of output messages.

Master Terminal Status

The following text is displayed for the primary master, secondary master, or system console:

```
PRIMARY MSTR
SECOND MSTR          ACTIVE , MESSAGES WAITING
SYS CONSOLE
```

ACTIVE indicates the terminal is in the following states:

```
OPERABLE
ALLOWING INPUT
ALLOWING OUTPUT
NOT STOPPED FOR QUEUEING
NOT LOCKED
NOT IN TEST MODE
```

MESSAGES WAITING indicates that there are system messages (Q3) waiting to be sent, thus preventing shutdown from completing.

OTMA PHASE=x

The current phase of IMS Open Transaction Manager Access (OTMA) processing. The shutdown phases are:

1 OTMA shutdown processing has begun.

2 Dependent regions have terminated. DFS1970 messages have been sent to OTMA clients for all inflight Send-then-Commit (commit mode 1) transactions.

3 OTMA is waiting for termination of all client-processing-related activities.

4 OTMA shutdown is complete.

COMMIT x TMEMBER=membername TPIPE=tpipename

The commit mode for the transaction and which OTMA client (member) and TPIPE is currently in progress and thus preventing completion of shutdown. COMMIT x can be either of the following:

0 Commit-Then-Send

1 Send-Then-Commit

XXXXXXX PROCESSING REGION(S) ACTIVE

Active regions, where XXXXXXX is one of the following types:

```
BMP MSG
TP  MSG
FP
```

If status of a line or link shows input or output in progress and immediate shutdown is desired, the /IDLE command can be used.

One of the following commands can be used to close the VTAM node:

- /CHECKPOINT(except simple checkpoint)
- /CLSDST
- /IDLE NODE(only after a /CLDST command has been tried first)
- /QUIESCE(for session type 6 nodes only)
- /STOP NODE

The /STOP DC or /CHECKPOINT(other than simple checkpoint) command can be issued to close the VTAM ACB.

If a checkpoint purge is in progress and messages are in the output queues, an orderly shutdown can be initiated by entering the /CLSDST or /STOP command. This allows termination after the next output message is sent, but might be insufficient for display terminals or terminals in input mode. If a node is hung, the following can be done:

1. A /CLSDST FORCE command can be issued.
2. If that fails, then the /IDLE command can be issued.

If any messages are queued for PTERM1 (system console) or the master terminal, they must be requested and received before the purge will complete. /STOP and /IDLE are ineffective against these terminals. When using ISC parallel sessions, one line of information is displayed for each applicable session of the node. The user name is also displayed.

/DISPLAY STATUS (With No Keywords)

/DISPLAY STATUS (with no keywords) displays the IMS resources that are in the states listed in Table 54. /DISPLAY STATUS displays only conditions that require operator intervention. Status definitions are shown in Appendix G, “Status and Attributes for the /DISPLAY Command,” on page 855.

Table 54. /DISPLAY STATUS Resource States Displayed (No Keywords)

Resource	State
CLASS	STOPPED
DATABASE	ALLOCF, BACKOUT, EEQE, INQONLY, LOCK, NOTINIT, NOTOPEN
DATABASE ADS	PRE-OPEN FAILED, SEVERE-ERROR, UNAVAILABLE, COPY-PHASE, FORMAT-PHASE
DATABASE AREA	RECOVERY NEEDED, STOPPED
LINE	IDLE, NOIN, NOOUT, NOQUEUE, NOTOPEN, PSTOPPED, PUR, RESP, RESP-INP, STOPPED
LINK	ACTV, COLD, ERE, IDLE, NOTIDLE-A, NOTIDLE-B, NOTIDLE-Cxx, NOTOPEN, NRE, N/A, PSTOPPED
LTERM/MSNAME	LOCK, PSTOPPED, PUR, QERROR, QLOCK, STOPPED
LUNAME	STO, STO-INP, STO-OUTP
NODE	CLSDST, CONV-HLD, C1INOP, C2INOP, C3INOP, C4INOP, DEACTIVATED, DISCONNECTED, FORCE, INOP, LOCK, LOST, OPNDST, QUIESCED, RESP, RESP-INP, RESYNC, SHUT, STOPPED
PROGRAM	DB-STOPD, I/O PREVENT, LOCK, NOTINIT, STOPPED

Table 54. /DISPLAY STATUS Resource States Displayed (No Keywords) (continued)

Resource	State
PTERM	COMPINOP, CONV-HLD, EXCL, INOP, LOCK, LOOPTEST, NOIN, NOTOPEN, NOOUT, NOQUEUE, PSTOPPED, PUR, RESP, RESP-INP, STOPPED, TEST
RTCODE	STOPPED
TMEMBER	STO
TRAN	LOCK, PSTOPPED, PUR, QERROR, SPND, STOPPED, USTOPPED
USER	CONV-HLD, DEADQ, RESP, RESP-INP, STOPPED

Any combination of the indicated keywords can be used to display the associated resources. If no exceptional conditions are found for any of the resources specified by the individual keywords, STATUS UNRESTRICTED is displayed.

For Fast Path databases, the /DISPLAY STATUS DATABASE command shows the area and ADS exceptions as well as the database status. Each area with exceptions is displayed on a separate line after the database line. Each ADS with exceptions is displayed on a separate line after the area line.

The /DISPLAY STATUS USER command provides the status of a user. The ALLOC status is followed by the nodename to which the user is allocated or signed onto in parenthesis.

In an IMSplex, if NODE, LTERM, or USER is specified with the /DISPLAY STATUS command, it will result in extensive accesses to the Resource Manager for global information, and their use should be carefully considered.

In a DBCTL environment, /DISPLAY STATUS with no keywords only displays database and program information.

In a DCCTL environment, /DISPLAY STATUS with no keywords does not display database information.

For /DISPLAY STATUS TRANSACTION, DFSCPIC is displayed as the PSB name for CPI Communications driven transactions that have not issued a DL/I APSB call to allocate a PSB.

For a DB/DC RSR tracking subsystem, /DISPLAY STATUS with no keywords only displays DATABASE, LINE, LTERM, NODE, PTERM, and USER information.

The /DISPLAY STATUS DATABASE and /DISPLAY STATUS commands show the status of online forward recovery (OFR) for those databases in an RSR tracking subsystem that have OFR in progress.

/DISPLAY STATUS LUNAME can indicate that a command such as /STO LUNAME ALL INPUT or /STO LUNAME ALL OUTPUT was entered before the /DISPLAY that applies to all future LU 6.2 inbound or outbound conversations.

/DISPLAY STATUS TMEMBER displays the IMS Open Transaction Manager Access (OTMA) transaction pipes that are stopped. If a transaction pipe is not stopped, its status is UNRESTRICTED. An example of using the /DISPLAY STATUS TMEMBER command is shown in “Example 5 for /DISPLAY STATUS Command” on page 337.

/DISPLAY STRUCTURE

/DISPLAY STRUCTURE displays the status of one or more coupling facility list queue structures used by IMS. This command displays the queue structure name, type, and status.

The status conditions that can be displayed are:

AVAILABLE
The structure is available to be used by IMS.

CONNECTED
IMS is connected to the structure.

DISCONNECTED
IMS is not connected to the structure.

IN-OVERFLOW
The structure is in overflow mode.

REBLD-INPROG
A rebuild is in progress for the structure.

SHUTDOWN-STRCHKPT
A structure checkpoint will be taken by CQS for the structure during a CQS normal shutdown.

STRCHKPT-INPROG
A structure checkpoint is in progress for the structure.

UNAVAILABLE
The structure is not available to be used by IMS.

This command is only valid in a shared-queues environment.

An example of using the /DISPLAY STRUCTURE command is shown in “Example for /DISPLAY STRUCTURE Command” on page 338.

/DISPLAY SUBSYS

/DISPLAY SUBSYS is used to display information about an external subsystem. (The external subsystem is **not** a CCTL subsystem.) When used without the OASN keyword, the SUBSYS keyword displays the status of the connection between IMS and the external subsystem (not CCTL), as well as all application programs communicating with the external subsystem (not CCTL). The OASN keyword is used with the SUBSYS keyword to display all OASNs associated with the specified external subsystem. (The external subsystem is **not** a CCTL subsystem.) For more information, see “/DISPLAY OASN SUBSYS” on page 242.

Examples of using the /DISPLAY SUBSYS command are shown in:

- “Example 1 for /DISPLAY SUBSYS Command” on page 338
- “Example 2 for /DISPLAY SUBSYS Command” on page 338
- “Example for /DISPLAY OASN Command” on page 316

For status terms for the connection between IMS and the external subsystem, see “Subsystem Status Terms” on page 266. For status terms for the connection between an IMS application program and the external subsystem, see “Dependent Region Status Terms” on page 266.

Subsystem status terms can be generated by the /DISPLAY SUBSYS command. Dependent region status terms can be generated by the /DISPLAY SUBSYS or /DISPLAY ACTIVE command.

Subsystem Status Terms

CONN The IMS control region has successfully completed a host system IDENTIFY request to the external subsystem, making the two subsystems aware of each other's existence so they can begin a normal dialog.

NOT CONN
The external subsystem is in an idle state; that is, the external subsystem has not been the object of the /STOP SUBSYS command, or the external subsystem initialization exit indicated not to issue the IDENTIFY REQUEST (connect)

CONN IN PROGRESS
The connection process for the specified subsystem is in progress

STOPPED
The specified subsystem has been successfully stopped via the /STOP SUBSYS command and all region connections to the specified external subsystem have been terminated

STOP IN PROGRESS
The /STOP SUBSYS command is in progress. Before it successfully completes, all active connections to the specified subsystem from all IMS regions must be quiesced.

TERM IN PROGRESS
An internal termination of the subsystem connection is underway. This type of termination was instigated by IMS abnormal condition processing, an external subsystem exit, or the external subsystem (the error message The exact reason). IMS shutdown will not cause this condition.

INVALID SUBSYSTEM NAME = XXXX
The specified subsystem name has not been defined in the IMS subsystem PROCLIB member

SUBSYSTEM XXXX NOT DEFINED BUT RECOVERY OUTSTANDING
The specified subsystem name has not been defined to IMS in the external subsystem PROCLIB member, but IMS still has outstanding recovery elements from a previous execution when the indicated subsystem was known

The command recognition character will also be displayed for the external subsystem.

Dependent Region Status Terms

CONN An IMS dependent region has successfully completed a host system IDENTIFY request to the external subsystem as a result of an application having been scheduled into the dependent region. In an MPP, the application does not have to issue an external subsystem call to cause a connection.

CONN, ACTIVE
An IMS application program has established communication with an external subsystem (for example, has issued at least one external subsystem call). At this point a thread exists between the IMS application program and the external subsystem.

The absence of a PSB name for a thread An a connection to the external subsystem exists but an application program is not currently occupying the region. The presence or absence of an LTERM name indicates whether or not a region is message driven.

/DISPLAY SYSID TRANSACTION

The /DISPLAY SYSID TRANSACTION command displays the IDs of the local and remote systems associated with the transaction. Values displayed for the local and remote IDs can range from 1 to 2036.

An example of using the /DISPLAY SYSID TRANSACTION command is shown in “Example for /DISPLAY SYSID Command” on page 338.

/DISPLAY TIMEOVER

The /DISPLAY TIMEOVER command displays all nodes that have been waiting for a VTAM response for a longer time than specified in the /TRACE SET ON TIMEOUT command.

An example of using the /DISPLAY TIMEOVER command is shown in “Example for /DISPLAY TIMEOVER Command” on page 339.

time# is the time period in minutes; it is required. The number of minutes must be between 1 and 60.

If no nodes are found that have been waiting longer than time# minutes, then the message NO NODE WITH TIMEOVER time# FOUND is displayed. An error message is displayed and the command is rejected if the timeout trace facility failed during IMS initialization. User names are displayed for ISC nodes, but non-ISC nodes show N/A.

/DISPLAY TMEMBER

The /DISPLAY TMEMBER command displays the current transaction member status for IMS Open Transaction Manager Access (OTMA) clients and servers. This command displays the following:

- Each member in each XCF group

  When you issue /DISPLAY TMEMBER ALL, the server is always the first member displayed.
- The XCF status for each member

  The status can be one of the following:

  – ACTIVE
  – CREATED
  – FAILED
  – NOT DEFINED
  – QUIESCED
  – UNKNOWN

  If the server leaves the XCF group (for example, because of a /STOP OTMA command), then the XCF status is displayed as NOT DEFINED.
- The user status for each member

  The status can be one of the following:

  - ACCEPT TRAFFIC
  - DISCONNECTED
  - IN SLOWDOWN
  - SERVER
  - WAIT BID
  - WAIT RESPONSE

  either SERVER or ACCEPT TRAFFIC.
- The security status for each server

  The security status can be one of the following:

  ```
  NONE
  CHECK
  FULL
  PROFILE
  ```

  See Chapter 53, "/SECURE," on page 583 for more information on the security status.

This command also displays transaction pipe status for each member, including:
- Enqueue and dequeue counts
- Current queue count
- Current status

This command shows one of the following for the status of a transaction pipe:

DQF Dequeue request for OTMA REPresynch command failed.

REP IMS is waiting for an OTMA REPresynch command from the client.

REQ IMS is sending an OTMA REQresynch command to the client.

RSF Reset request failed for an OTMA REPresynch command.

STO The transaction pipe is stopped.

SYN The transaction pipe is being synchronized.

TBR IMS is waiting for an OTMA TBResynch command from the client.

TMP The transaction pipe is temporary.

TRA The transaction pipe is being traced.

Examples of using the /DISPLAY TMEMBER command are shown in:
- "Example 1 for /DISPLAY TMEMBER Command" on page 339
- "Example 2 for /DISPLAY TMEMBER Command" on page 339

QCNT

Specifies that global queue count information is to be displayed. If QCNT is not specified, local queue counts are displayed; when it is specified, no local queue counts are displayed.

This keyword is only valid in a shared-queues environment.

/DISPLAY TRACE

/DISPLAY TRACE displays the status and options of the current IMS traces. TRACE must be followed by one or more keywords or by the ALL parameter. There is no default.

Examples of using the /DISPLAY TRACE command are shown in:

- "Example 1 for /DISPLAY TRACE Command" on page 339
- "Example 2 for /DISPLAY TRACE Command" on page 340
- "Example 3 for /DISPLAY TRACE Command" on page 341
- "Example 4 for /DISPLAY TRACE Command" on page 341
- "Example 5 for /DISPLAY TRACE Command" on page 342
- "Example 6 for /DISPLAY TRACE Command" on page 342
- "Example 7 for /DISPLAY TRACE Command" on page 342
- "Example 8 for /DISPLAY TRACE Command" on page 342
- "Example 9 for /DISPLAY TRACE Command" on page 343
- "Example 10 for /DISPLAY TRACE Command" on page 343
- "Example 11 for /DISPLAY TRACE Command" on page 343
- "Example 12 for /DISPLAY TRACE Command" on page 343
- "Example 13 for /DISPLAY TRACE Command" on page 343
- "Example 14 for /DISPLAY TRACE Command" on page 344

This command can display the following information:

ALL In a DBCTL system, ALL displays only DB-related traces. It does not show DC-related traces such as LINE, LUNAME, LINK, NODE, TRANSACTION, TRAP, or TIMEOUT.

In a DCCTL system, ALL displays only DC-related traces. It does not show DB-related traces such as OSAMGTF or PI.

For a DB/DC RSR tracking subsystem, ALL displays only LINE, NODE, PSB, TABLE, and TRAP information.

The display shows which external data set is active and whether the OLDS is being traced for trace logging. The command also indicates if XTRC is usable and the status of the XTRC data sets.

EXIT Displays user exit tracing. Currently, only tracing for the DFSMSCE0 user exit is supported. Table 55 lists the display exit entry points.

Table 55. Display Exit Entry Points for Tracing

LRDI	LRTR	TR62	TRVT
LRIN	PRCH	TRBT	
LRLT	PRIS	TROT	

The display indicates the status of each exit entry point. If the status is ON, the trace is active for the selected entry point. If the status is OFF, the trace is inactive for the selected entry point. If the status is N/A, the specified DFSMSCE0 trace entry point does not exist in the IMS.

LINE The relative line number and line type of all traced lines.

LINK The link number and link partner for all logical links.

LUNAME
All LU 6.2 application programs that are being traced.

OSAMGTF
Whether the OSAM Buffer Trace facility is active or inactive.

MONITOR
The status of the IMS Monitor, and if active, the options specified for it.

NODE The VTAM node name and node type. If the node is defined for a parallel session, each traced user of that node is displayed. If the users are not active, an N/A will be displayed.

PI The status of the program isolation trace, and if active, the status of the options and the volume level.

PROGRAM
The traced program name.

PSB The PSB name and trace option for each traced PSB.

TABLE
The trace status, logging status, total number of entries in each trace table, and the volume level.

For a DL/I trace, the volume level is displayed in the subcategories.

For an RSR tracking subsystem, TABLE does not show information about the LUMI, QMGR, RETR, SCHD, SQTT, or SUBS table traces.

In an IMSplex, the OCMD, RM, and SCI table traces are included in the table traces displayed.

TCO Displays whether or not tracing is active for the Time Controlled Operation (TCO).

TIMEOUT
The status of the IMS VTAM I/O Timeout Detection Facility, including the specifications given when the /TRACE SET ... TIMEOUT command was issued. The /DISPLAY TRACE TIMEOUT command will show active or inactive status, the timeout value in minutes, and the action to be taken when timeout occurs (AUTO, MSG, or no action). Option and Timeout values will be displayed only when the facility is active.

TMEMBER
The IMS Open Transaction Manager Access (OTMA) transaction pipes that are currently being traced for the given OTMA client.

TRANSACTION
The name of the transaction and the applicable PSB name for each trace.

TRAP The trap that is currently set, either 1 or 2; that trap is enabled for MFS serviceability aids.

XTRC External trace status. Whether XTRC is usable and the status of the XTRC data sets are also displayed.

/DISPLAY TRACKING STATUS

In an RSR complex, /DISPLAY TRACKING STATUS displays the status of a tracking subsystem. /DISPLAY TRACKING STATUS is valid on active and RSR tracking subsystems.

Examples of using the /DISPLAY TRACKING STATUS command are shown in:

- "Example 1 for /DISPLAY TRACKING STATUS Command" on page 344
- "Example 2 for /DISPLAY TRACKING STATUS Command" on page 345

/DISPLAY TRACKING STATUS on an Active Subsystem

/DISPLAY TRACKING STATUS on an active subsystem displays the following:

- Subsystem IMSID
- The global service group (GSG)
- The service group (SG)
- The service group status
- The status of the conversation with log router

/DISPLAY TRACKING STATUS on an active subsystem shows the following information:

ACTIVE SUBSYSTEM
The IMSID, global service group, service group, and service group status of the active subsystem

IMSID The IMSID of the active subsystem.

GSG-NAME
The 1- to 8-character global service group (GSG) name. The global service group is the collection of all IMS subsystems in an RSR complex (such as online IMS, utilities, and so on) that access a particular set of databases. An active subsystem can only be defined as part of **one** global service group.

SG-NAME
The name of the service group (SG) of which the active subsystem is a part. The service group name is 1 to 8 characters long.

SG-STATUS
The service group status, which can be one of the following:

IDENTIFICATION IN PROGRESS
An the IMS logger is in the process of identifying to the Transport Manager Subsystem but has not yet completed.

IDENTIFIED
The IMS logger has identified to the Transport Manager Subsystem. A status of IDENTIFIED appears as long as the logger is connected or identified to the TMS. This status also appears after a /START SERVGRP command.

LOST The IMS logger's connection to the Transport Manager Subsystem is gone due to TMS or VTAM failure.

NO LOG ROUTER SG DEFINED
There is an error and logger has no knowledge of a log router with which to communicate. There is no Log Router Service Group defined to DBRC.

NOT IDENTIFIED
The IMS logger has never identified to the Transport Manager Subsystem (that is, is in its initial state) or the IMS logger has not been able to identify since the most recent /STOP SERVGRP command.

STOPPED
The /STOP SERVGRP command was issued. If there is an active conversation, /STOP SERVGRP causes it to be terminated.

At each OLDS switch, the logger no longer attempts to establish a conversation with the log router. Logs are no longer sent to the tracking subsystem. A /START SERVGRP command is required to undo the STOPPED condition.

ACTIVE'S CONVERSATIONS WITH LOG ROUTERS
Status of the conversations between the active and tracking subsystems.

SG-NAME
The name of the service group (SG) of which the tracking subsystem is a part.

RECORDS-NOT-SENT
The number of log records that have not yet been sent to the tracker. The number of records is a number in the form of nnnK, where K represents 1024 records. The number is rounded off to the nearest K. If the number of records exceeds 999K, then >999K is displayed.

This field applies only if the conversation status is SENDING LOGS. Otherwise, N/A is displayed.

CONVERSATION STATUS
The conversation status, which can be one of the following:

ACCEPT IN PROGRESS
A log router is in the process of accepting a conversation request, but has not yet allocated the conversation.

ALLOCATION IN PROGRESS
Indicates a conversation is in the process of being allocated.

DEALLOCATION IN PROGRESS
A conversation is in the process of being deallocated.

NO ACTIVE CONVERSATION
Indicates there is no active conversation between the logger and the log router. Although there is no conversation, the logger does know about the log router.

At each OLDS switch, the logger will attempt to establish a conversation with the log router until it is successful. A /START SERVGRP command can be issued to cause the logger to attempt to establish a conversation with the log router before the next OLDS switch.

SENDING LOGS
The logger has an active conversation with the log router and is sending logs to the tracking site.

SUSPENDED LOGS
The logger has an active conversation with the log router but has suspended sending logs to the tracking site due to resource shortage. The conversation is still intact.

/DISPLAY TRACKING STATUS on a Tracking Subsystem

When issued on a tracking subsystem, `/DISPLAY TRACKING STATUS` displays the following:

- Subsystem IMSID
- Readiness level
- Global service group (GSG)
- Service group (SG)
- Service group status
- Tracking status
- Tracking activity
- Gaps
- Isolated log sender status
- Online forward recovery (OFR) status
- Current and restart milestone indexes
- DL/I tracking status
- Fast Path tracking status

`/DISPLAY TRACKING STATUS` shows the same information for an IMS DB/DC tracking subsystem or a DBCTL tracking subsystem.

`/DISPLAY TRACKING STATUS` on a tracker shows the following information:

TRACKING SUBSYSTEM
The following tracking subsystem information is displayed:

IMSID The IMSID of the tracking subsystem. The IMSID is 1 to 4 characters long.

READINESS-LEVEL
The readiness level of the tracking subsystem, which is either RECOVERY or DATABASE.

GSG-NAME
The global service group (GSG) name. The GSG is the collection of all IMS subsystems in an RSR complex, such as online IMS, utilities, and so on, which access a particular set of databases. A tracking subsystem tracks only **one** global service group. The global service group name is 1 to 8 characters long.

SG-NAME
The 1- to 8-character service group (SG) name.

SG-STATUS
The service group status, which can be IDENTIFIED, NOT IDENTIFIED, or STOPPED.

IDENTIFIED
The tracking subsystem has identified to the Transport Manager Subsystem (TMS), and can establish conversations with active loggers and isolated log sender (ILS) instances. IDENTIFIED appears as long as the tracking subsystem is connected or identified to the TMS.

NOT IDENTIFIED
The tracking subsystem has not attempted to identify to the TMS. NOT IDENTIFIED is the initial state.

STOPPED
The tracking subsystem is not identified to the TMS. This status can be the result of any of the following: a /STOP SERVGRP command entered from the tracking subsystem, an unsuccessful attempt to identify, or termination of the TMS.

TRACKING STATUS
The status of the tracking subsystem, which can be one of the following:

INACTIVE
The tracking subsystem is idle. The tracking subsystem has completed all tracking work and has received no more log records from the active subsystems it tracks.

For example, a status of INACTIVE appears if the active subsystems have shut down and the tracking subsystem has completed its tracking work.

TRACKING
The tracking subsystem is tracking one or more active subsystems and has tracking work to do.

PLANNED TAKEOVER REQUESTED
One of the active subsystems being tracked issued an /RTAKEOVER command, notified the tracker, and shut down. As soon as all of the active systems in the GSG have shut down, all log gaps have been filled, and log routing has completed, the tracking subsystem will shut down for the planned remote takeover.

PLANNED TAKEOVER IN PROGRESS
An /RTAKEOVER command was entered on the active subsystem and all of the active subsystems being tracked have successfully shut down. The tracking subsystem is in the process of shutting down for the planned remote takeover. The tracking subsystem must shut down before active subsystems can be started at the new active site.

UNPLANNED TAKEOVER IN PROGRESS
An /RTAKEOVER command was entered on the tracking subsystem. The tracking subsystem will shut down for the unplanned remote takeover after it has completed routing received log data.

NUMBER OF ACTIVE SUBSYSTEMS
The number of active subsystems currently being tracked. It is followed by an integer.

TRACKING ACTIVITY
Status of the conversations between the tracker and the loggers of the active subsystems being tracked. The following information is displayed for each active subsystem being tracked:

NAME The IMSID of the subsystem being tracked or the jobname for batch or batch backout being tracked. Subsystem names are 1 to 4 characters long. Job names are 1 to 8 characters long.

If the active subsystems are XRF-capable, the subsystem name displayed includes a hyphen and the one-character HSBID.

Example: The XRF alternate for IMSA would be displayed as IMSA-2.

TYPE Type of subsystem, which can be ONLINE, ACTIVE, BATCH, or BBO (batch backout), as described below:

ONLINE
An online (not batch) subsystem that has not sent data to the tracking subsystem. The subsystem might be an active subsystem that has not completed restart or it might be an XRF alternate subsystem.

ACTIVE
An online (not batch) subsystem that has sent data to the tracking subsystem.

BATCH
A batch subsystem that has sent data to the tracking subsystem.

BBO A batch backout job that has sent data to the tracking subsystem.

RECEIVED-LOG
The universal coordinated time of the last log record received from the subsystem, BATCH job, or BBO job. The timestamp is of the format `hhmmss`, where `hh` represents hours, `mm` represents minutes, and `ss` represents seconds. RECEIVED-LOG does not apply to conversations with XRF-alternate subsystems, in which case `N/A` is displayed.

ROUTED-LOG
The universal coordinated time of the last log routed by the log router. The timestamp is of the format `hhmmss`, where `hh` represents hours, `mm` represents minutes, and `ss` represents seconds. ROUTED-LOG does not apply to conversations with XRF-alternate subsystems, in which case `N/A` is displayed.

NOT-ROUTED
The number of log records that have been received by the tracker but not yet routed by the log router. The number is in the format `nnnK`, where `K` represents 1024 log records. The number is rounded off to the nearest `K`. If this number exceeds 999K, then `>999K` is displayed. NOT-ROUTED does not apply to conversations with XRF-alternate subsystems, in which case N/A is displayed.

STATUS
The tracking status. If a conversation exists between the tracking subsystem and the active IMS, the status field contains CONV-ACT. If there is no conversation, the status field is left blank.

GAPS The status of gaps between the tracker and the active subsystems, BATCH jobs, or BBO jobs. One line is displayed for each gap associated with an active subsystem.

If the gap information is not known to the tracking subsystem, the PRILOG token associated with the gap is displayed as “MISSING LOG, PRILOG TOKEN=”. The following gap information is displayed:

NAME The IMSID of the active system for which there is a gap, or the jobname of the BATCH or BBO job for which there is a gap.

In an XRF environment, the `NAME` is the RSENAME rather than the IMSID.

LOG-SEQ-FIRST
The log sequence number, in hexadecimal, of the first log record in the gap

IN-GAP
The number of records, in decimal, that are in the gap. The number is of the form nnnK, where K represents 1024 records. The number is rounded off to the nearest K. If this number exceeds 999K, then >999K is displayed.

RECEIVED
The number of records, in decimal, that have been received. The number is of the form nnnK, where K represents 1024 records. The number is rounded off to the nearest K. If this number exceeds 999K, then >999K is displayed.

NOT-ROUTED
The number of log records, in decimal, that have not been routed for the oldest gap. The number is of the form nnnK, where K represents 1024 records. If this number exceeds 999K, then >999K is displayed. This number does not apply to gaps other than the oldest one.

STATUS
The status of the gap, which is one of the following:

FILLING
The gap is currently being filled.

WAITING
The gap is not being filled yet. For example, the gap might be waiting to be filled if there are not enough conversations to send the log records, or a batch job is still running.

PERMNT ERR
There is a permanent error, such as a read error, that prevents this gap from being filled.

UNAVAIL
The log data needed to fill the gap is not available at the active site. The log or log data sets might have been deleted.

ISOLATED LOG SENDER STATUS
Status of isolated log sender conversations.

SYSTEM NAME OF TRANSPORT MANAGER SUBSYSTEM
If the tracking subsystem has a conversation with an isolated log sender, the system name of the Transport Manager to which the isolated log sender is identified is displayed.

NUMBER OF CONVERSATIONS
Number of conversations with this instance of isolated log sender.

ONLINE FORWARD RECOVERY STATUS
The following online forward recovery (OFR) status is displayed:

NUMBER OF STREAMS BEING PROCESSED
Number of OFR processes initiated. The online forward recoveries are processed one at a time.

OFR ROUTING TIME
The latest close time of any log data set that has been or is currently being read for OFR.

MILESTONE INDEX STATUS
The following milestone information is displayed:

CURRENT MILESTONE INDEX
The current milestone index.

RESTART MILESTONE INDEX
The restart milestone index.

DLI TRACKING STATUS
Status of DL/I database tracking.

TRACKING PSTs
Number of DL/I database tracking PSTs currently in use. The number can be from 0 to 255.

USAGE OF TRACKING PSTs
Average percent usage of DLI database tracking PSTs currently in use

BACKLOG OF REDO RECORDS IN DATASPACE
The backlog number of redo records in the data space. The backlog is a measure of how well the tracker is keeping up with the workload. The backlog is a number in the form of `nnnK`, where `K` represents 1024 redo records. The number is rounded off to the nearest `K`. If the number of redo records exceeds 999K, then `>999K` is displayed.

FP TRACKING STATUS
Status of Fast Path database tracking.

TRACKING PSTs
Number of FP database tracking PSTs and the reason the PSTs are invoked. The number of tracking PSTs can be from 0 to 255. The reason the PSTs are invoked can be one of the following and only appears if there are one or more PSTs:

AREAWRITE
The PSTs are invoked by a write request for a specific area.

MILESTONE
The PSTs are invoked by milestone processing.

THRESHOLD
The PSTs are invoked by threshold write.

CURRENT LOG VOLUME IN DATASPACE
Current log volume in Fast Path tracking data space. This number can be from 0 to `nnnnnnnK`, where `K` represents 1024 log records. The maximum possible number is 2000000K (two gigabytes). The number is rounded off to the nearest `K`.

MAX LOG VOLUME IN DATASPACE
Maximum log volume in Fast Path tracking data space. This number can be from 0 to `nnnnnnnK`, where `K` represents 1024 log records. The maximum possible number is 2000000K (two gigabytes). The number is rounded off to the nearest `K`.

/DISPLAY TRANSACTION

The /DISPLAY TRANSACTION command displays the class, priorities, queue counts, the number and size of output segments for a specified transaction code, and the number of regions the transaction is currently scheduled in, all on the first line. The next two indented lines show the PSB name associated with the transaction and the status. DFSCPIC is displayed as PSB name for CPI Communications driven transactions that have not issued a DL/I APSB call to allocate a PSB. Fast Path exclusive transaction codes only have the transaction code and class displayed. The queue counts are not shown because Fast Path processing bypasses the IMS message queues.

IMS Open Transaction Manager Access (OTMA) supports an architected format for the output from the /DISPLAY TRANSACTION command. The architected output is returned to the OTMA client, not to an IMS operator. See .

The TRANSACTION keyword parameter can be generic.

The status displayed can be one of the following:

```
BAL
DYNAMIC
I/O PREV
LOCK
PSTOP
PUR
QERR
SPND
STOP
TRA
USTOP
```

for an explanation of the status.

/DISPLAY TRANSACTION also displays I/O prevention when a BMP program containing GSAM cannot complete scheduling.

Examples of using /DISPLAY TRANSACTION are shown in “Example 1 for /DISPLAY TRANSACTION Command” on page 346 and “Example 2 for /DISPLAY TRANSACTION Command” on page 346.

QCNT

Specifies that global queue count information is to be displayed. If QCNT is not specified, local queue counts are displayed; when it is specified, no local queue counts are displayed.

IMS does not check that the specified name is a valid transaction in the IMS subsystem on which the command is entered.

The command /DISPLAY TRAN ALL QCNT displays all of the transactions on the shared queues with a global queue count. /DISPLAY TRAN ALL QCNT also displays an AFFINITY column which gives the IMS SYSID for transactions which have an affinity for a particular IMS system. This is valuable information particularly when there are multiple output lines for the same transaction name. See “Example 4 for /DISPLAY TRANSACTION Command” on page 347.

This keyword is only valid in a shared-queues environment.

/DISPLAY UOR

The /DISPLAY UOR command displays status information about IMS units of recovery (UORs) for protected resources on the RRS/MVS recovery platform.

If you specify the six-byte pseudo-token of the protected UOR (the *prtkn*), this command displays information about the requested UOR.

ACTIVE
Displays only active UORs.

ALL
Displays active and indoubt UORs. ALL is the default.

INDOUBT
Displays only UORs that were indoubt when they were terminated. Indoubt UORs identify residual work that needs resolution.

The output from the /DISPLAY UOR command consists of the following:

ST Status of the UOR, which can be one of the following:

- **A** Active; work is inflight.
- **B** Active; work is being backed out.
- **C** Active; work is being committed.
- **E** Active; work is in cleanup phase.
- **I** Active; work is indoubt, awaiting phase 2 action.
- **RI** Residual indoubt; indoubt UOR from a prior IMS execution or a dependent-region abend.

P-TOKEN
The IMS pseudo token, which can be used in subsequent commands to refer to this particular UOR. This token only displays for indoubt UORs (status RI). Use this token for subsequent /DISPLAY UOR and /CHANGE UOR commands.

PSBNAME
The IMS PSB scheduled for this UOR.

RRS-URID
The unique identifier for a protected unit of recovery (of which IMS work is part). RRS generates and maintains this data.

IMS-RECTOKN
The IMS recovery token that uniquely identifies the IMS portion of the UOR.

LUWID
The identifier of a work request across multiple distributed systems connected by LU 6.2 conversations. This data is only provided for /DIS UOR ACTIVE commands, and appears on a separate line of output.

EID The identifier of a work request across multiple connected distributed systems coordinated by the Encina toolkit. This data is only provided for /DIS UOR ACTIVE commands, and appears on a separate line of output.

XID X/Open identifier (XID). One possible work identifier for a distributed transaction used by a communications resource manager that uses X/Open distributed transaction processing model.

/DISPLAY USER

The /DISPLAY USER command displays all the USER structures and USERIDs that match the parameter or attribute specified.

The USER parameter can be generic when the generic parameter specifies users that already exist.

The following information is displayed:

- The USER field shows the USER structure name. N/A appears for all static USERIDs signed on to static nodes.
- The USERID field The RACF USERID that was used to sign on to the node. N/A appears if a USER structure is not associated with a node. In that case, the USER structure exists to hold status or has messages inserted to it. If no messages or status exist, the USER structure is deleted at the next checkpoint.
- The node name appears if the user is allocated to a node or signed on to a node.
- The cumulative queue counts for enqueues, dequeues, and number of items still on the queue for the specified users is displayed. The cumulative counts include all LTERMs associated with the specified users, whether the LTERMs are allocated to a node or not.

  For non-ISC static users, queue counts are displayed as N/A, since non-ISC static users are not associated with queues.
- STOPPED or ALLOC is displayed to show whether the user LTERMs are assigned to a node (ALLOC) or the user is stopped (STO).

USERIDs signed onto static nodes can only be used as parameters with the /DISPLAY USER command and the /SIGN ON command. USERIDs cannot be used as parameters with other commands that support the USER keyword.

“Attributes” are reserved parameters for the /DISPLAY USER command and cannot be used to name users. Any one attribute can be used with the USER keyword. Users are selected for display based on the attribute or attributes specified. The attribute display format is the same as the standard USER display. Attributes usually correspond to the conditions displayed for users.

Following is a list of the attributes that can be specified with the USER keyword:

- ALLOC
- CONVACT
- CONVHLD
- DEADQ
- EXCL
- MFST
- PRST
- RESP
- RESPINP
- STATIC
- STOPPED
- TEST

Explanations of the attributes can be found in Appendix G, “Status and Attributes for the /DISPLAY Command,” on page 855.

The ALLOC status is followed by the nodename (in parentheses) to which the user is allocated or signed on to.

The use of DEADQ with the /DISPLAY USER command shows all users that have queues marked as dead letter queues. When all the messages are purged by the /DEQUEUE command, the DEADQ status is removed. After all the messages have been purged and the status removed, then the user structure is deleted at the next simple checkpoint.

Examples of using the /DISPLAY USER command are shown in:

- “Example 1 for /DISPLAY USER Command” on page 348
- “Example 2 for /DISPLAY USER Command” on page 349
- “Example 3 for /DISPLAY USER Command” on page 349
- “Example 4 for /DISPLAY USER RECOVERY Command” on page 350

AUTOLOGON
Displays the current autologon information for the specified user. The autologon information can be changed dynamically with the /CHANGE command.

EMHQ
Specifies that global queue count information on the Expedited Message Handler Queues (EMHQ) displays when both EMHQ and QCNT are specified. If EMHQ is specified, QCNT must also be specified. This keyword is only valid in a shared-queues environment.

QCNT
Specifies that global queue count information displays. If QCNT is not specified, local queue counts are displayed; when it is specified, no local queue counts are displayed. If QCNT is specified and EMHQ is not, global queue count information on the shared message queues (MSGQ) displays.

IMS does not check if the specified name is a valid LTERM in the IMS subsystem on which the command is entered.

This command does not display queue counts for Fast Path output messages for the NODE unless EMHQ and QCNT are specified.

This keyword is only valid in a shared-queues environment.

RECOVERY
The /DISPLAY NODE|USER RECOVERY command can be used to display the recovery values that pertain to the node or user. The recovery values may be displayed from the local control blocks, if available, or from values saved for the node or user in the Resource Manager, if resource information is being kept in Resource Manager, and the node or user is defined to have its status kept in Resource Manager. The recovery information displayed with this command are:

- IMS owner

  The IMS ID (RSEname if an XRF system) of the IMS system that currently owns this resource. If the resource is not currently owned, this field will be displayed as NONE.
- Status Recovery Mode (SRM)

  The scope of recovery for a resource, and from where the end-user significant status is maintained and recovered.
- End-user significant status

The following status are end-user significant status for nodes and users.

– Conversation
– STSN
– Fast Path

These are status that frequently change for a resource, and thus there are performance considerations related to maintaining end-user significant status. The installation can tell IMS how to recover end-user significant status by specifying the level of recovery for each status. Shown for each resource is the level of recovery for each end-user significant status.

Examples

The following are examples of the /DISPLAY command.

Examples for /DISPLAY ACTIVE Command

The following are examples of the /DISPLAY ACTIVE command.

Example 1 for /DISPLAY ACTIVE Command

Entry ET:

```
/DISPLAY ACTIVE
```

Response ET:

```
  REGID JOBNAME   TYPE  TRAN/STEP PROGRAM  STATUS           CLASS
        MSGRGN    TP    NONE
        BATCHREG  BMP   NONE
        FPRGN     FP    NONE
        DBTRGN    DBT   NONE
      1 IMSMPP0   TPE   DSN8PS    DSN8IP13                  1,  3
        BATCHREG  BMP   NONE
        FPRGN     FP    NONE
        DBR3CTA3  DBRC
        DLI3CTA3  DLS
  VTAM ACB OPEN                   -LOGONS ENABLED
  LINE ACTIVE-IN -    1 ACTIV-OUT -    0
  NODE ACTIVE-IN -    0 ACTIV-OUT -    4
  LINK ACTIVE-IN -    0 ACTIV-OUT -    0
  *89041/163619*
```

Example 2 for /DISPLAY ACTIVE Command

Entry ET:

```
  /DISPLAY A
```

Response ET:

```
 REGID JOBNAME    TYPE  TRAN/STEP PROGRAM  STATUS          CLASS
     1 I3YMPP     TP                       WAITING          4,  1,  2,  3
       BATCHREG   BMP   NONE
       FPRGN      FP    NONE
       DBRC       DBRC
 VTAM ACB CLOSED
 LINE ACTIVE-IN -   1 ACTIV-OUT -    0
 NODE ACTIVE-IN -   0 ACTIV-OUT -    0
 *89041/100117*
```

Example 3 for /DISPLAY ACTIVE Command

Entry ET:

```
  /DISPLAY A
```

Response ET:

```
REGID JOBNAME   TYPE  TRAN/STEP PROGRAM  STATUS          CLASS
    2 I431MPP   TP    SKS7      DFSDDLT7                  1,  3,  6,  7
    3 IMS1BMP   BMP   REGION    SIS04P01
    1 I431IFP   FPM   TXCDRN07  DDLTRN07
      R3        DBRC
VTAM ACB OPEN                -LOGONS ENABLED
LINE ACTIVE-IN -   2 ACTIV-OUT -   0
NODE ACTIVE-IN -   0 ACTIV-OUT -   0
LINK ACTIVE-IN -   0 ACTIV-OUT -   0
*89041/144425*  DFSRSENM  ACTIVE
```

Explanation: The system is XRF capable so the date time stamp includes the RSE name and system indicator.

Example 4 for /DISPLAY ACTIVE Command

Entry ET:

```
/DISPLAY A
```

Response ET:

```
REGID JOBNAME   TYPE  TRAN/STEP PROGRAM  STATUS            CLASS
    1 I431MPP   TP    TXCDRN18  DDLTRN18 WAIT-INPUT          1,  3,  6,  7
      BATCHREG  BMP   NONE
    2 I431IFP   FPM   NO MSG.   DDLTRN07
      R3        DBRC
VTAM ACB OPEN                -LOGONS ENABLED
LINE ACTIVE-IN -   1 ACTIV-OUT -   0
NODE ACTIVE-IN -   0 ACTIV-OUT -   0
LINE ACTIVE-IN -   0 ACTIV-OUT -   0
*89041/132348*  DFSRSENM  ACTIVE
```

Explanation: Transaction TXCDRN18 is waiting for an input message. Program DDLTRN07 currently has no messages to process. Region 1 is in WAIT-FOR-INPUT (WFI) mode.

Example 5 for /DISPLAY ACTIVE Command

Entry ET:

```
/DISPLAY A
```

Response ET:

```
REGID JOBNAME   TYPE  TRAN/STEP PROGRAM  STATUS          CLASS
    2 I431MPP   TP    SKS7      DFSDDLT7 WAIT-MESSAGE    1,  3,  6,  7
    3 IMS1BMP   BMP   REGION    SIS04P01
    1 I431IFP   FPM   TXCDRN07  DDLTRN07
      R3        DBRC
VTAM ACB OPEN                -LOGONS ENABLED
LINE ACTIVE-IN -   2 ACTIV-OUT -   0
NODE ACTIVE-IN -   0 ACTIV-OUT -   0
LINK ACTIVE-IN -   0 ACTIV-OUT -   0
*89041/144425*  DFSRSENM  ACTIVE
```

Explanation: Transaction SKS7 is waiting for an input message. Program DFSDDLT7 currently has no messages to process. Region 2 is in Pseudo Wait-For-Input (PWFI) mode.

Example 6 for /DISPLAY ACTIVE Command

Entry ET:

```
/DISPLAY A
```

Response ET:

```
** SUBSYSTEM NAME **
      SYS3
REGID JOBNAME TYPE TRAN/STEP PROGRAM  STATUS       CLASS
    2 I431MPP TP   SKS7      DFSDDLT7 WAIT-MESSAGE 1,3,6,7
    3 IMS1BMP BMP  REGION    SIS04P01
    1 I431IFP FPM  TXCDRN07  DDLTRN07
      R3      DBRC
VTAM ACB OPEN          -LOGONS ENABLED
LINE ACTIVE-IN-    2 ACTIV-OUT-    0
NODE ACTIVE-IN-    0 ACTIV-OUT-    0
LINK ACTIVE-IN-    0 ACTIV-OUT-    0
*89041/144425*  DFSRSENM  ACTIVE
```

Explanation: Transaction SKS7 is waiting for an input message. Program DFSDDLT7 currently has no messages to process. Region 2 is in Pseudo Wait-For-Input (WPFI) mode.

Example 7 for /DISPLAY ACTIVE Command

Entry ET:

```
/DISPLAY ACTIVE
```

Response ET:

```
REGID JOBNAME   TYPE  TRAN/STEP PROGRAM  STATUS           CLASS
    1 IMSMPPA   TPI   TPN1      DFSCPIC  WAIT-BLOCKMOVER   20, 22, 24, 26
    2 IMSMPPB   TPI   TPN2      DFSCPIC                    21, 23, 25, 22
    3 IMSMPPC   TP    TPN3      SAAPSB1                    28, 29, 24, 23
    4 IMSMPPD   TPI   TPN4      DFSCPIC  WAIT-SYNCPOINT    14, 15, 16, 17
    4 IMSMPPE   TP    TRAN1     LU2PGMA  WAIT-INPUT         2,  4,  6,  8
    5 IMSMPPF   TP    TRAN2     LU2PGMB                    1,  3,  5,  7
VTAM ACB OPEN          / APPC LUNAME=IMSLUNME ENABLED -LOGONS ENABLED
LINE ACTIVE-IN -    2 ACTIV-OUT  -    0
NODE ACTIVE-IN -    0 ACTIV-OUT  -    0
LINK ACTIVE-IN -    0 ACTIV-OUT  -    0
*90332/114253*
```

Example 8 for /DISPLAY ACTIVE Command

Entry ET:

```
/DISPLAY ACTIVE
```

Response ET:

```
REGID JOBNAME   TYPE  TRAN/STEP PROGRAM  STATUS      CLASS
      MSGRGN    TP    NONE
      BATCHREG  BMP   NONE
      FPRGN     FP    NONE
      DBTRGN    DBT   NONE
      DBRSCHA5  DBRC
      DLISCHA5  DLS
VTAM ACB OPEN          -LOGONS DISABLED
IMSLU=N/A.N/A            APPC STATUS=DISABLED  TIMEOUT=   0
APPLID=APPL8     USERVAR=IMS1
LINE ACTIVE-IN -    1 ACTIV-OUT -    0
NODE ACTIVE-IN -    0 ACTIV-OUT -    0
LINK ACTIVE-IN -    0 ACTIV-OUT -    0
*95080/183050*  DFSRSENM  ACTIVE
```

Explanation: The IMS subsystem is both XRF-capable and RSR-capable.

Example 9 for /DISPLAY ACTIVE Command

Entry ET:

```
/DISPLAY ACTIVE
```

Response ET:

```
REGID JOBNAME   TYPE  TRAN/STEP PROGRAM  STATUS      CLASS
    1 IMSMPPA   TP                       WAITING       4,  1,  2
    2 IMS1BMP   BMP   REGION    SIS04P01 WAIT-AOI
    4 IMS2BMP   BMP   REGION    SIS04P02 WAIT-AOI
    3 I510MPP   TP                       WAIT-EPCB POOL   8,   8,  8,  8
    2 IMS3BMP   BMP                      WAIT-EPCB POOL

      FPRGN     FP    NONE
      DBRC      DBRC
VTAM ACB CLOSED
LINE ACTIVE-IN -   1 ACTIV-OUT -    0
NODE ACTIVE-IN -   0 ACTIV-OUT -    0
*96280/095345*
```

Example 10 for /DISPLAY ACTIVE Command

Entry ET:

```
/DISPLAY ACTIVE DC
```

Response ET:

```
VTAM STATUS AND ACTIVE DC COUNTS
VTAM ACB OPEN                -LOGONS ENABLED
LINE ACTIVE-IN -   1 ACTIV-OUT -   0
NODE ACTIVE-IN -   1 ACTIV-OUT -   0
LINK ACTIVE-IN -   1 ACTIV-OUT -   0
*89041/110704*
```

Example 11 for /DISPLAY ACTIVE Command

Entry ET:

```
/DISPLAY ACTIVE DC
```

Response ET:

```
VTAM STATUS AND ACTIVE DC COUNTS
VTAM ACB OPEN                  -LOGONS ENABLED
IMSLU=SYS1.IMSLUNME      STATUS=ENABLED
APPLID=APPL8     USERVAR=IMS1
LINE ACTIVE-IN -    2 ACTIV-OUT -    0
NODE ACTIVE-IN -    0 ACTIV-OUT -    0
LINK ACTIVE-IN -    0 ACTIV-OUT -    0
*92232/114253*
```

3 ## Example 12 for /DISPLAY ACTIVE Command

3 Entry ET:

3
```
/DISPLAY ACTIVE DC
```

3 Response ET:

3
```
VTAM STATUS AND ACTIVE DC COUNTS
VTAM ACB CLOSED                      -APPLID ACB
VTAM ACB OPEN        -LOGONS ENABLED  -MNPS ACB
IMSLU=N/A.N/A    APPC STATUS=DISABLED  TIMEOUT=   0
OTMA GROUP=N/A        STATUS=NOTACTIVE
APPC/OTMA SHARED QUEUE STATUS - LOCAL=INACTIVE  GLOBAL=INACTIVE
APPC/OTMA RRS MAX TCBS - 2 ATTACHED TCBS - QUEUED SENDS -  0  N
APPLID=APPL8     GRSNAME=         STATUS=DISABLED MNPS=APPL1
LINE ACTIVE-IN -    1 ACTIV-OUT -    0
NODE ACTIVE-IN -    0 ACTIV-OUT -    0
LINK ACTIVE-IN -    0 ACTIV-OUT -    0
*03069/143346*  DFSRSENM  ACTIVE
```

2 Explanation: An XRF system is using MNPS. The MNPS ACB is APPL1 which is
2 open and is accepting logons. The APPLID ACB is APPL8, but it is closed.

Example 13 for /DISPLAY ACTIVE Command

Entry ET:

```
/DISPLAY ACTIVE REGION
```

Response ET:

```
REGID JOBNAME   TYPE  TRAN/STEP PROGRAM  STATUS  CLASS
      MSGRGN    TP    NONE
      BATCHREG  BMP   NONE
      FPRGN     FP    NONE
      DBRGCTAN  DBRC  NONE
*89041/102934*
```

Example for /DISPLAY AFFINITY Command

Entry ET:

```
/DISPLAY AFFIN NODE NDSLU2A1
```

Response ET:

```
NODE     APPLID
NDSLU2A1 APPL9
*97098/162106*
```

Explanation: The node has an affinity for APPL ID APPL9.

Entry ET:

```
/DISPLAY AFFIN NODE LU37722
```

Response ET:

```
NODE     APPLID
LU37722 N/A
*97098/162156*
```

Explanation: The node has no affinities.

Example for /DISPLAY AOITOKEN Command

Entry ET:

```
/DISPLAY AOITOKEN
```

Response ET:

```
AOITOKEN      ENQCT        QCT       SEGS W-REGID
AOITOKN1          4          2         12    NONE
AOITOKN2          0          0          0     2,4
AOITOKN3          1          0          0    NONE
AOITOKN4 1000000000 1000000000 2147483647    NONE
*92280/095345*
```

Example for /DISPLAY APPC Command

Entry ET:

```
/DISPLAY APPC
```

Response ET:

```
IMSLU    #APPC-CONV SECURITY STATUS   DESIRED
IMSLUNME          0 FULL     ENABLED  ENABLED
*90347/114253*
```

Entry ET:

```
  /DISPLAY APPC
```

Response ET:

```
  IMSLU    #APPC-CONV SECURITY STATUS   DESIRED
  IMSLUNME          0 FULL     DISABLED ENABLED
  *90347/114253*
```

```
Entry ET:
  /DISPLAY APPC

Response ET:

IMSLU             #APPC-CONV SECURITY STATUS   DESIRED
SYS1.IMSLUNME              0 FULL     STOPPED  STOPPED
*92232/114253*
```

Examples for /DISPLAY AREA Command

The following are examples of the /DISPLAY AREA command.

Example 1 for /DISPLAY AREA Command

Entry ET:

```
  /DIS AREA db21ar1 db21ar3 db21ar6 IOVF
```

Response ET:

```
  AREANAME  EQECT   TOTAL UNUSED  TOTAL UNUSED      DBNAME EEQECT CONDITIONS
   DDNAME   REMAIN SEQ DEPENDENT DIR ADDRESSABLE
  DB21AR1     N/A      13     11     74     74-CI DEDBJN21 999    VIR, PREO, PREL
   DB21AR1     10     N/A    N/A    N/A    N/A        N/A
   DB21AR1B    10     N/A    N/A    N/A    N/A        N/A
   DB21AR1C    10     N/A    N/A    N/A    N/A        N/A
  DB21AR3     N/A      13     11     74     56-CI DEDBJN21        PREO
   DB21AR3     10     N/A    N/A    N/A    N/A        N/A
   DB21AR3B    10     N/A    N/A    N/A    N/A        N/A
  DB21AR6     N/A      17     15     74     56-CI DEDBJN21
   DB21AR6     10     N/A    N/A    N/A    N/A        N/A
  *93076/173254*
```

Explanation: DIR ADDRESSABLE means Independent Overflow part.

Example 2 for /DISPLAY AREA Command

Entry ET:

```
/DIS AREA ALL
```

Response ET:

```
  AREANAME  EQECT   TOTAL UNUSED  TOTAL UNUSED      DBNAME EEQECT CONDITIONS
   DDNAME   REMAIN SEQ DEPENDENT DIR ADDRESSABLE

  DB21AR0     N/A      17     15     74    N/A    DEDBJN21 999    VIR, PREO, PREL
   DB21AR0     10     N/A    N/A    N/A    N/A        N/A
  DB21AR1     N/A      13     11     74    N/A    DEDBJN21        VIR, PREO
   DB21AR1     10     N/A    N/A    N/A    N/A        N/A
   DB21AR1B    10     N/A    N/A    N/A    N/A        N/A
   DB21AR1C    10     N/A    N/A    N/A    N/A        N/A
  DB21AR10    N/A      17     15     74    N/A    DEDBJN21        VIR, PREO, PREL
   DB21AR10    10     N/A    N/A    N/A    N/A        N/A
  DB21AR11    N/A     133    131     74    N/A    DEDBJN21        VIR
   DB21AR11    10     N/A    N/A    N/A    N/A        N/A
  DB21AR2     N/A      19     17     74     0-UW DEDBJN21        VIR, PREO, PREL
  UTILITY:  HSRE  PVTTOT#     45 PVTAVL#    15
   DB21AR2     10     N/A    N/A    N/A    N/A        N/A
  DB21AR3     N/A      13     11     74    N/A    DEDBJN21        PREO
```

```
 DB21AR3     10     N/A    N/A    N/A    N/A        N/A
 DB21AR3B    10     N/A    N/A    N/A    N/A        N/A
DB21AR4     N/A     N/A    N/A    N/A    N/A    DEDBJN21        STOPPED, NOTOPEN
DB21AR5     N/A     N/A    N/A    N/A    N/A    DEDBJN21        STOPPED, NOTOPEN
DB21AR6     N/A     N/A    N/A    N/A    N/A    DEDBJN21        STOPPED, NOTOPEN
DB21AR7     N/A     N/A    N/A    N/A    N/A    DEDBJN21        NOTOPEN
DB21AR8     N/A     N/A    N/A    N/A    N/A    DEDBJN21        NOTOPEN
DB21AR9     N/A     N/A    N/A    N/A    N/A    DEDBJN21        NOTOPEN
*93076/173254*
```

Explanation: DIR ADDRESSABLE means Independent Overflow part.

Example 3 for /DISPLAY AREA Command

Entry ET:

```
/DIS AREA db11ar4
```

Response ET:

```
AREANAME  EQECT   TOTAL UNUSED   TOTAL UNUSED      DBNAME EQECT CONDITIONS
 DDNAME   REMAIN  SEQ   DEPENDENT DIR ADDRESSABLE

DB21AR4    N/A     N/A    N/A    N/A    N/A    DEDBJN21   999   STOPPED, NOTOPEN,
                                                               RECOVERY
  *97136/173254*
```

Explanation: RECOVERY means this Fast Path area is undergoing recovery with the Online Recovery Service.

Example 4 for /DISPLAY AREA

Entry ET:

```
/DIS AREA DB21AR0
```

Response ET:

```
DFS000I  AREANAME  EQECT   TOTAL UNUSED  TOTAL UNUSED  DBNAME   EEQECT  CONDITIONS   SYS3
DFS000I  DDNAME    REMAIN  SEQ  DEPENDENT DIR ADDRESSABLE                             SYS3
DFS000I  DB21AR0   N/A     29    28      74    0-UW    DEDBJN21 0   VIR, PREO, PREL  SYS3
DFS000I  UTILITY:  HSSP  PVTTOT#   45 PVTAVL#    30    IMSID: IMS2                   SYS3
DFS000I  DB21AR01    10     N/A    N/A    N/A    N/A        N/A                      SYS3
DFS000I  DB21AR02    10     N/A    N/A    N/A    N/A        N/A                      SYS3
DFS000I  DB21AR03    10     N/A    N/A    N/A    N/A        N/A                      SYS3
```

Explanation: The IMS Fast Path utility, HSSP, is active on IMSID=IMS2.

Examples for /DISPLAY ASSIGNMENT Command

The following are examples of the /DISPLAY ASSIGNMENT command.

Example 1 for /DISPLAY ASSIGNMENT Command

Entry ET:

```
/DISPLAY ASSIGNMENT LINE 2 PTERM ALL
```

Response ET:

```
TERMINAL
  2- 1  IN  - L2740S2
        OUT - L2740S1, L2740S2
*90295/130910*
```

Entry ET:

```
/DISPLAY ASSIGNMENT LINE 4 PTERM 1, 2
```

Response ET:

```
TERMINAL
  4- 1  IN - L2740SM1
        OUT- L2740SM1
  4- 2  IN - L2740SM2
        OUT- L2740SM2
*90295/130328*
```

Entry ET:

```
/DISPLAY ASSIGNMENT LINK ALL
```

Response ET:

```
LINK PLINK    SIDR SIDL MSNAME
   1 BLCB1CTC   1    4  SYSTEM2A
                1    5  SYSTEM2B
                3    6  FRESNO
   2 BLCB2BSC  21   23  BOSTON
               22   24  ALBANY
   3 BLCB1MTM  19   20  TEST1
*90280/072935*
```

Entry ET:

```
/DISPLAY ASSIGNMENT LTERM DIALQ1 INQUIRY1 L2740S1 MASTER WTOR
```

Response ET:

```
LTERM    IN-TERMINAL  OUT-TERMINAL  USER
DIALQ1      11-   1     11-   1
INQUIRY1     5-   1      5-   1
L2740S1      3-   1      3-   1
MASTER       3-   1      3-   1
WTOR         1-   SC     1-   SC
*90295/130657*
```

Entry ET:

```
/DISPLAY ASMT LTERM L1 L2 L3
```

Response ET:

```
LTERM    IN-TERMINAL  OUT-TERMINAL  USER
L1          14-   1     14-   1
L2        NODEA   -1  NODEA   -1    A
L3        N/A     -1  N/A     -1    B
*90231/143059*
```

Entry ET:

```
/DISPLAY ASSIGNMENT LTERM ALL
```

Response ET:

```
LTERM    IN-TERMINAL  OUT-TERMINAL  USER
CRDPUNCH    14-   1     14-   1
DIALQ1      11-   1     11-   1
DIALQ2      12-   1     12-   1
DIALQ3      13-   1     13-   1
INQUIRY1     5-   1      5-   1
INQUIRY2     6-   1      6-   1
INQUIRY3     7-   1      7-   1
INQUIRY4     8-   1      8-   1
INQUIRY5     9-   1      9-   1
INQUIRY6    10-   1     10-   1
L2740SM1     4-   1      4-   1
L2740SM2     4-   2      4-   2
L2740S1      3-   1      3-   1
L2740S2      2-   1      2-   1
```

```
MASTER       3-   1      3-   1
PTPPUNCH    14-   1     14-   1
WS12DS    WS12-   1   WS12-   1
WTOR         1-   SC     1-   SC
*90295/160953*
```

Entry ET:

```
/DISPLAY ASSIGNMENT LTERM ALL
```

Response ET:

```
LTERM    IN-TERMINAL  OUT-TERMINAL  USER
CRDPUNCH    14-   1     14-   1
DIALQ1      11-   1     11-   1
DIALQ2      12-   1     12-   1
DIALQ3      13-   1     13-   1
INQUIRY1     5-   1      5-   1
INQUIRY2     6-   1      6-   1
INQUIRY3     7-   1      7-   1
INQUIRY4     8-   1      8-   1
INQUIRY5     9-   1      9-   1
INQUIRY6    10-   1     10-   1
L2740SM1     4-   1      4-   1
L2740SM2     4-   2      4-   2
L2740S1      3-   1      3-   1
L2740S2      2-   1      2-   1
MASTER       3-   1      3-   1
PTPPUNCH    14-   1     14-   1
WS12DS    WS12-   1   WS12-   1
WTOR         1-   SC     1-   SC
*90295/160953*
```

Entry ET:

```
/DISPLAY ASSIGNMENT MSNAME ALL
```

Response ET:

```
LINK PLINK    SIDR SIDL MSNAME   IMS1
     N/A        32    3 LINK23B4   IMS1
  10 PLNK12V    20   10 LINK12V1   IMS1
     N/A        35    3 LINK23U2   IMS1
     N/A        28    3 ELINK211   IMS1
   1 PLNK12V    31   11 LINK13B3   IMS1
  10 PLNK12V   266 2036 LINK13V    IMS1
  10 PLNK12V   256 1012 LINK12V    IMS1
   2 PLNK12C    33   13 LINK13C2   IMS1
  10 PLNK12V    80  300 LINK13X    IMS1
  11 PLNK12V    40    9 LINK12V2   IMS1
   3 PLNK12M    34   14 LINK13M2   IMS1
  12 PLNK13V    38   18 LINK13V2   IMS1
   7 N/A        43   44 ELINK124   IMS1
     N/A        36    3 LINK23V1   IMS1
   1 PLNK12V    21   11 LINK12B1   IMS1
   1 PLNK12V    22   12 LINK12B2   IMS1
  10 PLNK12V    90  500 LINK12Y    IMS1
   5 N/A        26   16 ELINK120   IMS1
   1 PLNK12V    32   12 LINK13B4   IMS1
     N/A         8    3 LINK21V3   IMS1
   6 N/A        51   41 ELINK1MS   IMS1
   3 PLNK12M    24   14 LINK12M1   IMS1
     N/A       266    3 LINK23V    IMS1
   4 N/A        35   15 LINK13U2   IMS1
     N/A        31    3 LINK23B3   IMS1
     N/A        43    3 ELINK224   IMS1
     N/A        80    3 LINK23X    IMS1
     N/A        37    3 LINK23V2   IMS1
```

```
  6 N/A        27   17 ELINK121   IMS1
    N/A        33    3 LINK23C2   IMS1
  4 N/A        25   15 LINK12U1   IMS1
    N/A        34    3 LINK23M2   IMS1
  5 N/A        29   19 ELINK122   IMS1
    N/A        29    3 ELINK222   IMS1
  2 PLNK12C    23   13 LINK12C1   IMS1
 10 PLNK12V    36   16 LINK13V1   IMS1
  7 N/A        42   45 ELINK123   IMS1
 16 LKV1TSTB   93    3 PTH3TSTB   IMS1
*97258/170244*
```

Entry ET:

```
/DISPLAY ASSIGNMENT MSPLINK ALCB3MTM
```

Response ET:

```
LINK PLINK    TYPE  ADDR     MAXSESS  NODE
   3 ALCB3MTM MTM   ****
*90280/072754*
```

Entry ET:

```
/DISPLAY ASMT NODE NLU1 NLUT6S NLUT6P
```

Response ET:

```
NODE     USER
NLU1              IN -L1      , L2
                  OUT-L2      , L1
NLUT6S            IN -L3
                  OUT-L3
NLUT6P   A        IN -L5      , L6
NLUT6P   N/A      IN -NONE
                  OUT-NONE
NLUT6P   C        IN -L7
                  OUT-L7
*90229/083059*
```

Entry ET:

```
/DISPLAY ASMT NODE NLUT6P USER C
```

Response ET:

```
NODE     USER
NLUT6P   C        IN -L7
                  OUT-L7
*90229/083059*
```

Entry ET:

```
/DISPLAY ASSIGNMENT NODE ALL
```

Response ET:

```
NODE     USER
CT3275            IN -VT3275  , VT3275P
                  OUT-VT3275  , VT3275P
CT3277A           IN -VT3270A
                  OUT-VT3270A
CT3277B           IN -VT3270B
                  OUT-VT3270B
CT3277C           IN -VT3270C
                  OUT-VT3270C
CT3277D           IN -VT3270P1
                  OUT-VT3270P1
CT3277E           IN -VT3270P2
```

```
                  OUT-VT3270P2
L3270A            IN -VT3270L1
                  OUT-VT3270L1
L3270B            IN -VT3270L2
                  OUT-VT3270L2
L3284A            IN -VT3270P3
                  OUT-VT3270P3
W612              IN -W610JP  , W610D6  , W610FP
                  OUT-W610D6  , W610FP  , W610JP
W613              IN -W620JP  , W620D6  , W620PB  , W620M6
                  OUT-W620D6  , W620JP  , W620M6  , W620PB
FCBOX1            IN -W6CIT
                  OUT-W6CIT
*90295/091334*
```

Entry ET:

```
/DISPLAY ASMT USER A B C D
```

Response ET:

```
USER     ID       NODE
A        CICSA    NLUT6P   I/O- L1      , L2
B                          I/O- L3
C        CICSB    NLUT6P   I/O- L4
D                          I/O- NONE
*90229/083059*
```

Entry ET:

```
/DISPLAY ASSIGNMENT SYSID ALL
```

Response ET:

```
LINK PLINK    SIDR SIDL MSNAME
   1 BLCB1CTC   1    4  SYSTEM2A
   1 BLCB1CTC   2    5  SYSTEM2B
   1 BLCB1CTC   3    6  FRESNO
   3 BLCB1MTM  19   20  TEST1
   2 BLCB2BSC  21   23  BOSTON
   2 BLCB2BSC  22   24  ALBANY
*90280/072821*
```

Entry ET:

```
/DISPLAY ASSIGNMENT SYSID ALL
```

Response ET:

```
LINK PLINK    SIDR SIDL MSNAME   IMS1
     N/A         8    3 LINK21V3   IMS1
     N/A        28    3 ELINK211   IMS1
   5 N/A        29   19 ELINK122   IMS1
   1 PLNK12V    31   11 LINK13B3   IMS1
   1 PLNK12V    32   12 LINK13B4   IMS1
   2 PLNK12C    33   13 LINK13C2   IMS1
   3 PLNK12M    34   14 LINK13M2   IMS1
   4 N/A        35   15 LINK13U2   IMS1
  10 PLNK12V    36   16 LINK13V1   IMS1
     N/A        37    3 LINK23V2   IMS1
  12 PLNK13V    38   18 LINK13V2   IMS1
   7 N/A        43   44 ELINK124   IMS1
   6 N/A        51   41 ELINK1MS   IMS1
  10 PLNK12V    80  300 LINK13X    IMS1
  10 PLNK12V   266 2036 LINK13V    IMS1
*97258/170653*
```

Entry ET:

```
  /DISPLAY ASSIGNMENT SYSID 22 23 24
```

Response ET:

```
  LINK PLINK    SIDR SIDL MSNAME
     2 BLCB2BSC  22   24  ALBANY
    23     IS LOCAL
    24     IS LOCAL
  *90280/072902*
```

Example 2 for /DISPLAY ASSIGNMENT Command

Entry ET:

```
/DIS ASMT USER IMSUS01 IMSUS02
```

Response ET:

```
USER     USERID   ID       NODE
IMSUS01  IMSUS01           DTSLU201 I/O- IMSUS01
N/A      IMSUS01           L3270A   I/O- T3270LA
N/A      IMSUS02           L3270B   I/O- T3270LB
DT327002 IMSUS02           DT327002 I/O- DT327002
*92350/112229*
```

Entry ET:

```
/DIS ASMT USER IMSUS01*
```

Response ET:

```
USER     USERID   ID       NODE
IMSUS01C N/A                        I/O- NONE
IMSUS01D N/A                        I/O- NONE
IMSUS01  IMSUS01           DTSLU201 I/O- IMSUS01
N/A      IMSUS01           L3270A   I/O- T3270LA
*92350/113904*
```

Entry ET:

```
/DIS ASMT USER ALL
```

Response ET:

```
USER     USERID   ID       NODE
CA13                                I/O- CA1SP3T1, CA1SP3T2
IMSUS01C N/A                        I/O- NONE
IMSUS01D N/A                        I/O- NONE
LU6SPA   N/A               LU6NDPF  I/O- LU6LTPA1
LU6SPB   N/A                        I/O- LU6LTPB1, LU6LTPB2
DT327002 IMSUS02           DT327002 I/O- DT327002
IMSUS03  IMSUS03           DTSLU601 I/O- IMSUS03
IMSUS01  IMSUS01           DTSLU201 I/O- IMSUS01
IMSUS05  N/A                        I/O- NONE
N/A      IMSUS02           L3270B   I/O- T3270LB
N/A      IMSUS01           L3270A   I/O- T3270LA
N/A      IMSUS04           L3270C   I/O- T3270LC
*92350/114845*
```

Examples for /DISPLAY CCTL Command

The following are examples of the /DISPLAY CCTL command.

Example 1 for /DISPLAY CCTL Command

Entry ET:

```
  /DISPLAY CCTL
```

Response ET:

```
CCTL              STATUS
CICS1
*89067/114857*
```

Example 2 for /DISPLAY CCTL Command

Entry ET:

```
/DISPLAY CCTL ALL
```

Response ET:

```
CCTL     PSEUDO-RTKN RECOVERY-TOKEN   REGID  PSBNAME   STATUS
CICS1                                                   ATTACHED
                     0000000000000000     3            AVAILABLE
                     0000000000000000     2            AVAILABLE
                     0000000000000000     1            AVAILABLE
 *89067/113236*
```

Example 3 for /DISPLAY CCTL Command

Entry ET:

```
/DISPLAY CCTL CICS1
```

Response ET:

```
CCTL     PSEUDO-RTKN RECOVERY-TOKEN   REGID  PSBNAME   STATUS
CICS1                                                   ATTACHED
                     9FFA9568FF594301     3  BMP255    ACTIVE
                     9FFA956B7AE24E00     2  BMP255    ACTIVE
                     0000000000000000     1            AVAILABLE
 *89067/113251*
```

Example 4 for /DISPLAY CCTL Command

Entry ET:

```
/DISPLAY CCTL CICS1 ACTIVE
```

Response ET:

```
CCTL     PSEUDO-RTKN RECOVERY-TOKEN   REGID  PSBNAME   STATUS
CICS1                                                   ATTACHED
                     9FFA967C26D69802     2  BMP255    ACTIVE
                     9FFA967ACF9EB802     1  BMP255    ACTIVE
 *89067/114557*
```

Example 5 for /DISPLAY CCTL Command

Entry ET:

```
/DISPLAY CCTL CICS1 INDOUBT
```

Response ET:

```
CCTL     PSEUDO-RTKN RECOVERY-TOKEN   REGID  PSBNAME   STATUS
CICS1                                                   ATTACHED
         000100C0    9FFA956B7AE24E00        BMP255    INDOUBT
         00010040    9FFA9568FF594301        BMP255    INDOUBT
 *89067/113413*
```

Example for /DISPLAY CONVERSATION Command

Entry ET:

```
/DISPLAY CONVERSATION
```

Response ET:

```
TERMINAL USER   ID STATUS
NO CONVERSATIONS
*91113/182917*
```

Example for /DISPLAY CONVERSATION Command

Entry ET:

```
/DISPLAY CONVERSATION
```

Response ET:

```
TERMINAL USER   ID STATUS
11-   2          0001 HELD
 4-   2          0002 ACTIVE, SCHEDULED
11-   4          0001 HELD
 2-   1          0001 HELD
 5-   1          0002 ACTIVE
*91115/135706*
```

Entry ET:

```
/DISPLAY CONVERSATION
```

Response ET:

```
TERMINAL USER    ID STATUS
LUNAME1  USERID1  0001 ACTIVE
DT327001 IMSUS01  0005 ACTIVE
LUNAME2           0002 ACTIVE,SCHEDULED
LUNAME3  USERID4  0003 ACTIVE,SCHEDULED
LUNAME1  USERID4  0004 SCHEDULED
*90332/114253*
```

Entry ET:

```
/DISPLAY CONVERSATION BUSY
```

Response ET:

```
TERMINAL USER   ID STATUS
NO BUSY CONVERSATIONS
*91113/182959*
```

Entry ET:

```
/DISPLAY CONVERSATION BUSY
```

Response ET:

```
TERMINAL USER   ID STATUS
 2-   1          0001 ACTIVE, SCHEDULED
 4-   2          0010 ACTIVE, SCHEDULED
 4-   1          0011 ACTIVE, SCHEDULED
*91113/192021*
```

Entry ET:

```
/DISPLAY CONVERSATION BUSY LINE 4 PTERM ALL
```

Response ET:

```
TERMINAL USER   ID STATUS
 4-   2          0001 ACTIVE, SCHEDULED
 4-   1          0002 ACTIVE, SCHEDULED
*91113/192101*
```

Entry ET:

```
/DISPLAY CONVERSATION BUSY
```

Response ET:

```
TERMINAL           USER        ID STATUS
NETWORK1.LUNAME1   USERID1   0001 ACTIVE
DT327001           IMSUS01   0005 ACTIVE
```

```
NID2.LUNAME2                0002 ACTIVE,SCHEDULED
LUNAME3           USERID4   0003 ACTIVE,SCHEDULED
NETWORK1.LUNAME1  USERID5   0004 ACTIVE
*92232/114253*
```

Entry ET:

```
/DISPLAY CONVERSATION HELD
```

Response ET:

```
TERMINAL USER   ID STATUS
NO HELD CONVERSATIONS
*91113/183022*
```

Entry ET:

```
/DISPLAY CONVERSATION HELD NODE NSLUTP6
```

Response ET:

```
TERMINAL USER   ID STATUS
NSLUTP6  SPOOLA  0007   HELD
NSLUTP6  SPOOLC  000A   HELD
*91113/183033*
```

Example for /DISPLAY CQS Command

Entry ET:

```
/DISPLAY CQS
```

Response ET:

```
JOBNAME    VERS#  STATUS
CQS1       1.1    CONNECTED
*95200/170817*
```

Explanation: IMS is connected to the Common Queue Server, CQS1.

Examples for /DISPLAY DATABASE Command

The following are examples of the /DISPLAY DATABASE command.

Example 1 for /DISPLAY DATABASE Command

Entry ET:

```
/DISPLAY DATABASE ALL
```

Response ET:

```
DATABASE  TYPE  TOTAL UNUSED  TOTAL UNUSED ACC  CONDITIONS
DD41M702  DL/I                             EX   NOTOPEN
DD41M803  DL/I                             EX   NOTOPEN
DEDBJN21  DEDB    SEQ DEPEND DIRECT ADDRES EX   NOTOPEN
DB21AR0   AREA    N/A    N/A    N/A    N/A      NOTOPEN
DB21AR1   AREA    N/A    N/A    N/A    N/A      NOTOPEN
DB21AR2   AREA    N/A    N/A    N/A    N/A      NOTOPEN
DB21AR3   AREA    N/A    N/A    N/A    N/A      NOTOPEN
DB21AR4   AREA    N/A    N/A    N/A    N/A      NOTOPEN
DB21AR5   AREA    N/A    N/A    N/A    N/A      NOTOPEN
DB21AR6   AREA    N/A    N/A    N/A    N/A      NOTOPEN
DB21AR7   AREA    N/A    N/A    N/A    N/A      NOTOPEN
DB21AR8   AREA    N/A    N/A    N/A    N/A      NOTOPEN
DB21AR9   AREA    N/A    N/A    N/A    N/A      NOTOPEN
DB21AR10  AREA    N/A    N/A    N/A    N/A      NOTOPEN
DB21AR11  AREA    N/A    N/A    N/A    N/A      NOTOPEN
DEDBJN22  DEDB    SEQ DEPEND DIRECT ADDRES EX   NOTOPEN
DB22AR0   AREA    N/A    N/A    N/A    N/A      NOTOPEN
```

```
DB22AR1   AREA    N/A    N/A    N/A    N/A      NOTOPEN
DEDBJN23  DEDB    SEQ DEPEND DIRECT ADDRES EX   NOTOPEN
DB23AR0   AREA    N/A    N/A    N/A    N/A      NOTOPEN
DB23AR1   AREA    N/A    N/A    N/A    N/A      NOTOPEN
DIMSRNO1  DL/I                             EX   NOTOPEN
DIMSRNO2  DL/I                             EX   NOTOPEN
DIMSRNO3  DL/I                             EX   NOTOPEN
*89184/142639*
```

Entry ET:

```
/DISPLAY DB DD41M803 BKERR
```

Response ET:

```
DATABASE
DD41M803
    ERROR DD   TYPE     BLOCK
    DD41M803   IOT      0000003F
*90135/161902*
```

Explanation: An I/O toleration error queue element exists for database DD41M803 on a newly created active system in an XRF environment.

Following an /UNLOCK SYSTEM command on the above system, the /DISPLAY DB DD41M803 BKERR command would yield the following display:

```
DATABASE
DD41M803
    NO EEQE OR INCOMPLETE BACKOUT INFORMATION AVAILABLE
*90135/163500*
```

Example 2 for /DISPLAY DATABASE Command

Entry ET:

```
/DIS DB BE3ORDER BE3PARTS.
```

Response ET:

```
DATABASE  TYPE  TOTAL UNUSED  TOTAL UNUSED ACC  CONDITIONS
BE3ORDER  DL/I                             EX   STOPPED, LOCK, NOTOPEN,
                                                RECALL
BE3PARTS  DL/I                             EX   NOTOPEN, RECALL
*94277/124039*
```

Example 3 for /DISPLAY DATABASE Command

Entry ET:

```
/DIS DB RECALL.
```

Response ET:

```
DATABASE  TYPE  TOTAL UNUSED  TOTAL UNUSED ACC  CONDITIONS
BE3ORDER  DL/I                             EX   STOPPED, LOCK, NOTOPEN,
                                                RECALL
BE3PARTS  DL/I                             EX   NOTOPEN, RECALL
IVPDB1    DL/I                             UP   STOPPED, LOCK, NOTOPEN,
                                                RECALL
IVPDB2    DL/I                             UP   NOTOPEN, RECALL
*94277/124119*
```

Example 4 for /DISPLAY DATABASE Command

Entry ET:

```
/DISPLAY DATABASE DEDBJN21 BKERR
```

Response ET:

```
DATABASE
DEDBJN21
    ERROR DD  TYPE    BLOCK
    DB21AR0   IOT     00015000
    DB21AR0   IOT/VSO 00000001
    DB21AR1   IOT     00054000
    DB21AR1   IOT/VSO 00000001
```

Explanation: The /DIS DATABASE command with the BKERR keyword does not display individual RBAs for I/O tolerated non-SDEP CIs of a DEDB area defined with the Virtual Storage Option (VSO). Instead, a single entry with a value of 00000001 is displayed and the EEQE type is set to IOT/VSO. I/O tolerated SDEP CIs for the area are displayed individually by RBA.

Example 5 for /DISPLAY DATABASE Command

Entry ET:

```
/DISPLAY DATABASE OFR
```

Response ET:

```
DATABASE  TYPE  TOTAL UNUSED  TOTAL UNUSED ACC  CONDITIONS
BE2PCUST  DL/I                             EX   ALLOCS OFR
BE3ORDER  DL/I                             EX   ALLOCS OFR
BE3ORDRX  DL/I                             EX   ALLOCS OFR
BE3PARTS  DL/I                             EX   ALLOCS OFR
BE3PSID1  DL/I                             EX   ALLOCS OFR
*91240/132406*   SYS3
```

Example 6 for /DISPLAY DATABASE Command

If a full function database data set is undergoing recovery with Online Recovery Service, the output of the /DISPLAY DATABASE command includes this information in the command response.

Entry ET:

```
/DISPLAY DATABASE DD41M702
```

Response ET:

```
DATABASE  TYPE  TOTAL UNUSED  TOTAL UNUSED ACC  CONDITIONS
DD41M702  DL/I                             EX   STOPPED, LOCK, NOTOPEN
                                                RECOVERY
*97184/142639*
```

Example 7 for /DISPLAY DATABASE Command

Display the HALDB master for the PHIDAM database called PARTMAST.

Entry ET:

```
/DIS DB PARTMAST
```

Response ET:

```
DATABASE  TYPE    TOTAL UNUSED  TOTAL UNUSED ACC  CONDITIONS
PARTMAST  PHIDAM                           UP
PART1     PART                             UP   ALLOCS
PART2     PART                             UP   NOTOPEN
PART3     PART                             UP   STOPPED
*99166/092514*
```

Example 8 for /DISPLAY DATABASE Command

Display the HALDB partition for PHIDAM PART2.

Entry ET:

```
/DIS DB PART2
```

Response ET:

```
DFS000I   DATABASE  TYPE    TOTAL UNUSED  TOTAL UNUSED ACC  CONDITIONS
DFS000I   PARTMAST  PHIDAM                             UP
DFS000I   PART2     PART                               UP   NOTOPEN
*99166/092737*
```

Example for /DISPLAY DBD Command

Entry ET:

```
/DISPLAY DBD MSDBLM01 MSDBLM02 MSDBLM03 MSDBLM04
```

Response ET:

```
DBD-NAME    TYPE    PSB-NAME    ACCESS
MSDBLM01    MSDB    DDLTLM06    R/W
                    DDLTLM07    R/W
MSDBLM02    MSDB    DDLTLM06    R/W
                    DDLTLM07    R/W
MSDBLM03    MSDB    DDLTLM06    R/W
                    DDLTLM07    R/W
MSDBLM04    MSDB    DDLTLM06    R/W
                    DDLTLM07    R/W
*91068/110958*
```

Example for /DISPLAY DESCRIPTOR Command

Entry ET:

```
/DISPLAY DESCRIPTOR ALL
```

Response ET:

```
DESC     LUNAME   TPNAME            MODE    SIDE     SYNCLEVEL  TYPE
LU62DES1 LUNAME4  ACCOUNT           ALPHA   SIDENME1 CONFIRM    MAPPED
LU62DES2 LUNAME7  APPLE67890123456+ BETHA   SIDENME2 NONE       BASIC
                  789END
*90332/114253*
```

Entry ET:

```
/DISPLAY DESCRIPTOR ALL
```

Response ET:

```
DESC      LUNAME          MODE    SIDE      SYNCLEVEL TYPE
LU62DES1  NID2.LUNAME4    ALPHA   SIDENME1  CONFIRM   MAPPED
   TPNAME: ACCOUNT
LU62DES2  LUNAME7         BETHA   SIDENME2  NONE      BASIC
   TPNAME: APPLE67890123456789012345
*92232/114253*
```

Example for /DISPLAY FDR Command

Entry ET:

```
/DISPLAY FDR
```

Response ET:

```
FDR-REGION  GROUPNAME  TIMEOUT
ACTIVE      FDBRSYS3   055
*96232/114253*
```

Example for /DISPLAY FPVIRTUAL Command

Entry ET:

```
/DISPLAY FPVIRTUAL
```

Response ET:

```
DATASPACE MAXSIZE(4K) AREANAME AREASIZE(4K) OPTION   SYS3
     001       524188
                     DB21AR1            76 PREO
                     DB21AR11          152
     000       524188                      DREF
                     DB21AR0            19 PREO, PREL
                     DB21AR10           19 PREO, PREL
                     DB21AR2            38 PREO, PREL
*93076/173254*
```

Entry ET:

```
/DISPLAY FPVIRTUAL
```

Response ET:

```
   DATASPACE MAXSIZE(4K) AREANAME AREASIZE(4K) OPTION
        000       524263                      DREF
                        DB21AR4           167 PREO, PREL
                        DB21AR2            42 PREO, PREL
        001       524263
                        DB21AR3            84 PREO
                        DB21AR1            84
   CF     POOL  CACHE    AREANAME AREASIZE(4K) OPTION
 CF1      4K     N       AREAFR01         1000 PREO, PREL
 CF2      512    Y       AREA2             100 PREO, PREL
 *95225/173254*
```

Entry ET:

```
/DISPLAY FPVIRTUAL
```

2 Response ET:

```
DATASPACE MAXSIZE(4K) AREANAME AREASIZE(4K) OPTION     SYS3
      000     524238                        DREF       SYS3
                      DB21AR1            19 PREO, PREL
                      DB21AR0            19 PREO, PREL
                      DB23AR1            19 PREO, PREL
      001     524238                                   SYS3
NO AREAS LOADED INTO NORM DATASPACE 001. SYS3
      002     524238                                   SYS3
NO AREAS LOADED INTO NORM DATASPACE 002. SYS3
AREANAME STRUCTURE ENTRIES CHANGED AREA CI# POOLNAME OPTIONS   SYS2
NO BLOCK LEVEL SHARED AREAS EXIST.    SYS3
*03129/145840*  SYS3
```

| Entry ET:

```
/DISPLAY FPVIRTUAL
```

| Response ET:

```
AREANAME STRUCTURE            ENTRIES CHANGED AREACI#  POOLNAME OPTIONS
DD01AR0  DD01AR0STR1          0000045 0000000 1234567  POOL2    PREO, PREL
DB33FR01 DB33FR01STRUCTUR     0000550 0000327          POOL5    PREO
DB99AR12 DB99AR12STR1         0000150 0000000          1024     PREO
AAAAAAAA V$$$CFSTRUCTURE1     0000100 0000050          1024     PREO
BBBBBBBB V$$$CFSTRUCTURE1     0000100 0000050          1024     PREO, PREL
CCCCCCCC V$$$CFSTRUCTURE1     0000100 0000050          1024     PREO
DDDDDDDD V$$$CFSTRUCTURE1     0000100 0000050          1024     PREO, PREL
```

```
DB21AR1  V$$$CFSTRUCTURE2      0000010 0000005          1024     PREO
DB21AR2  V$$$CFSTRUCTURE2      0000090 0000010          1024     PREO
DB21AR3  V$$$CFSTRUCTURE2      0000200 0000050          1024     PREO, PREL
DB21AR4  V$$$CFSTRUCTURE2      0000050 0000010          1024     PREO
DB21AR5  V$$$CFSTRUCTURE2      0000150 0000010          1024     PREO
```

Explanation: Each area using a multi-area structure is listed individually with its own statistics. The column, AREACI#, displays the total number of CIs in the root addressable part. This is the total number of CIs loaded into the structure if the entire Area was preloaded.

Example for /DISPLAY HSB Command

Entry ET (Active System):

```
/DISPLAY HSB
```

Response ET:

```
RSENAME       STATUS     MODE     IMS-ID    VTAM UVAR
DFSRSENM      ACTIVE              IMSA      USERVAR1
SURVEILLANCE  INTERVAL   TIMEOUT  STATUS
LOG                 10        99  ACTIVE
LNK                  3         9  ACTIVE
RDS                 10        99  ACTIVE
*89340/094236*
```

3 Entry ET (Active System - MNPS environment):

```
3 /DISPLAY HSB
```

3 Response ET:

```
3 RSENAME       STATUS     MODE     IMS-ID    MNPS NAME
3 DFSRSENM      ACTIVE              IMSA      USERVAR1
3 SURVEILLANCE  INTERVAL   TIMEOUT  STATUS
3 LOG                 10        99  ACTIVE
3 LNK                  3         9  ACTIVE
3 RDS                 10        99  ACTIVE
3 *89340/094236*
```

Entry ET (Alternate System):

```
/DISPLAY HSB
```

Response ET:

```
RSENAME       STATUS    PHASE    IMS-ID    VTAM UVAR  ACT-ID    LOG-TIME
DFSRSENM      BACKUP    TRK      IMSB      USERVAR1   IMSA      09:42:56
              BACKUP SYSTEM                ACTIVE SYSTEM
SURVEILLANCE  INTERVAL  TIMEOUT  STATUS    INTERVAL   TIMEOUT   STATUS
LOG                 10       99  INACTIVE        10        99   INACTIVE
LNK                  3        9  INACTIVE         3         9   INACTIVE
RDS                 10       99  INACTIVE        10        99   INACTIVE
TAKEOVER CONDITIONS - ALARM
VTAM                          *RDS LOG
*89340/094256*
```

3 Entry ET (Alternate System - MNPS environment):

```
3 /DISPLAY HSB
```

3 Response ET:

```
3 RSENAME       STATUS    PHASE    IMS-ID    MNPS NAME  ACT-ID    LOG-TIME
3 DFSRSENM      BACKUP    TRK      IMSB      USERVAR1   IMSA      09:42:56
3               BACKUP SYSTEM                ACTIVE SYSTEM
3 SURVEILLANCE  INTERVAL  TIMEOUT  STATUS    INTERVAL   TIMEOUT   STATUS
```

```
3                    LOG                10       99  INACTIVE        10        99   INACTIVE
3                    LNK                 3        9  INACTIVE         3         9   INACTIVE
3                    RDS                10       99  INACTIVE        10        99   INACTIVE
3                    TAKEOVER CONDITIONS - ALARM
3                    VTAM                          *RDS LOG
3                    *89340/094256*
```

Example for /DISPLAY HSSP Command

Entry ET (Active System):

```
/DISPLAY HSSP
```

Response ET:

```
RGN TYP JOBNAME   PSB       DATABASE  AREAS-OPTIONS   SYS3
  1 BMP CSSP020B  BMPFPE02  DEDBJN21  IC=(1,C)   SYS3
*89122/094325*   SYS3
```

Example for /DISPLAY LINE Command

Entry ET:

```
/DISPLAY LINE 1, 3, 5, 10
```

Response ET:

```
LINE TYPE     ADDR    RECD ENQCT DEQCT   QCT  SENT
   1 CONSOLE  ****       0     1     0     1     6
   3 2740-II  ****       0     0     0     0     0 STOPPED IDLE
   5 3270 R    0C9      43    51    43     8    79
  *85098/141118*
```

Entry ET:

```
/DISPLAY LINE ALL
```

Response ET:

```
LINE TYPE     ADDR    RECD ENQCT DEQCT   QCT  SENT
   1 CONSOLE  ****       4     1     1     0     6
   2 2740-I   ****       0     0     0     0     0 IDLE NOTOPEN
   3 2740-II  ****       0     0     0     0     0 STOPPED IDLE
   4 2740 NSC ****       0     0     0     0     0 STOPPED IDLE
   5 3270 R    0C9      43    51    43     8    79
   6 3270 L   ****       0     0     0     0     0 STOPPED IDLE
   7 3270 L   1234      20    22    20     2     0
   8 SYSTEM/3 ****       0     0     0     0     0 STOPPED IDLE
   9 SYSTEM/7 ****       0     0     0     0     0 STOPPED IDLE
  10 2741     ****       0     0     0     0     0 STOPPED IDLE
  11 2741 SW  ****       0     0     0     0     0 STOPPED IDLE
  12 2741 SW  ****       0     0     0     0     0 STOPPED IDLE
  13 RDR/PTR  ****       0     0     0     0     0 STOPPED IDLE
  14 RDR/PTR  ****       0     0     0     0     0 STOPPED IDLE
  15 RDR/PTR  ****       0     0     0     0     0 STOPPED IDLE
  16 RDR/PTR  ****       0     0     0     0     0 STOPPED IDLE
  17 2740 SW  ****       0     0     0     0     0 STOPPED IDLE
  18 DIAL     POOL             0     0     0  POOL STOPPED IDLE
*85098/141244*
```

Entry ET:

```
/DISPLAY LINE 5 PTERM ALL
```

Response ET:

```
LIN/PTE  TYPE     ADDR   RECD ENQCT DEQCT   QCT  SENT
  5-   1 3270 R   404081    3     6     6     0     8
  5-   2 3270 R   C14081    0     6     6     0     7
```

```
   5-   3 3270 R   C1C181   40    30    28     2    61
   5-   4 3270 R   C1C281    0     3     0     3     0 COMPINOP
   5-   5 3270 R   C1C381    0     3     0     3     0 COMPINOP
   5-   6 3270 R   C1C481    0     3     3     0     3
  *85098/141412*
```

Entry ET:

```
/DISPLAY LINE PSTOPPED, PUR
```

Response ET:

```
 LINE     TYPE   ADDR   RECD ENQCT DEQCT   QCT  SENT
   4    3270 R   ****      4     8     8     0    33   PUR IDLE NOTOPEN
  18    3270 R   022       0     4     4     0     4   PSTOPPED IDLE
 *85098/141283*
```

Examples for /DISPLAY LINK Command

The following are example of the /DISPLAY LINK command.

Example 1 for /DISPLAY LINK Command

Entry ET:

```
/DISPLAY LINK ALL
```

Response ET:

```
 LINK PARTNER   RECD  ENQCT  DEQCT    QCT   SENT
    1 AB           0      0      0      0      0 PSTOPPED IDLE COLD
    2 AC           8     21     12      9     12 NOTIDLE-C0C ACTV TRA
    3 AD           4      6      6      0      6 IDLE ACTV TRA
    4 BC           0      0      0      0      0 PSTOPPED IDLE COLD
    5 AE           0      3      0      3      0 PSTOPPED IDLE
 *91293/171240*
```

Entry ET:

```
/DISPLAY LINK 1 3
```

Response ET:

```
 LINK PARTNER   RECD  ENQCT  DEQCT    QCT   SENT
    1 AB           0      1      0      1      0 IDLE
    3 AD           0      0      0      0      0 PSTOPPED IDLE
 *90280/072548*
```

Example 2 for /DISPLAY LINK Command

Entry ET:

```
/DISPLAY LINK ALL MODE
```

Response ET:

```
LINK PARTNER   DEF  MODETBL  ACT  MODETBL
   1 AB        N/A           N/A
   2 AC        N/A           N/A
   3 AD        N/A           N/A
   4 AE        N/A           N/A
   5 AK        MSC12V
   6 AL        MS12V
*90179/102238*
```

Explanation: N/A is displayed in the mode table name fields of non-VTAM MSC links. MSC12V and MS12V are mode table names defined at system definition or established using the /CHA command.

Examples for /DISPLAY LTERM Command

The following are examples of the /DISPLAY LTERM command.

Example 1 for /DISPLAY LTERM Command

Entry ET:

```
/DISPLAY LTERM DIALQ1 INQUIRY1 L2740S1 L2740SM1 MASTER WTOR
```

Response ET:

```
LTERM     ENQCT    DEQCT    QCT
DIALQ1        0        0      0
INQUIRY1      0        0      0
L2740S1       1        1      0  STOP
L2740SM1      1        0      1  STOP
MASTER       12       12      0
WTOR          0        0      0
*90295/123755*
```

Example 2 for /DISPLAY LTERM Command

Entry ET:

```
/DISPLAY LTERM ALL
```

Response ET:

```
LTERM     ENQCT    DEQCT    QCT
DSPLY1        0        0      0
DSPLY2        0        0      0
DSPLY3        0        0      0
MASTER        8        8      0
WTOR          0        0      0
2740AA1       0        0      0
2740AA2       0        0      0
2740AA3       0        0      0
2740C1        0        0      0
2740C2        0        0      0
2740SM1       2        2      0 STOP
2740SM2       2        2      0 STOP
2740S1        1        1      0
2740S2        1        1      0 QERR
LU6A          1        1      0 QLOCK
*91276/125448*
```

Example 3 for /DISPLAY LTERM Command

Entry ET:

```
/DISPLAY LTERM LTERMA LTERM123 ABCD QCNT
```

Response ET:

```
LTERM        GBLQCT
LTERMA   1000000000
LTERM123         25
ABCD              0
*95200/170817*
```

Example 4 for /DISPLAY LTERM Command

Entry ET:

```
/DISPLAY LTERM EMHQ QCNT
```

(one user is on the queue)

Response ET:

```
LTERM        GBLQCT
FPE0001           1
*98203/145857*
```

Example for /DISPLAY LUNAME Command

Entry ET:

```
/DISPLAY LUNAME luname1 luname2 INPUT
```

Response ET:

```
LUNAME   #APPC-CONV
LUNAME1          15 TRA
LUNAME2           5
*90332/114253*
```

Entry ET:

```
/DISPLAY LUNAME luname1 luname3 BADLUNME OUTPUT
```

Response ET:

```
LUNAME        ENQCT      DEQCT      QCT      CONVCT
LUNAME1           5          3        2           2
LUNAME3           5          4        1           1
BADLUNME IS INVALID
*90332/114253*
```

Entry ET:

```
/DISPLAY LUNAME LUNAME1 TPNAME TPNAME1 BADTPNME
```

Response ET:

```
LUNAME/TPNAME        ENQCT      DEQCT      QCT
LUNAME1
-TPNAME1                 0          0        0 STO
BADTPNME IS INVALID
*90332/114253*
```

Entry ET:

```
/DISPLAY LUNAME LUNAME1 TPNAME ALL OUTPUT
```

Response ET:

```
LUNAME/TPNAME        ENQCT      DEQCT      QCT
LUNAME1
-TPNAME1                 0          0        0 STO
-TPNAME7890123456+       5          3        2 TRA,STO
 789END
-DFSSIDE                 2          2        0
*90332/114253*
```

Entry ET:

```
/DISPLAY LUNAME LUNAME2 TPNAME ALL
```

Response ET:

```
LUNAME/TPNAME        ENQCT      DEQCT      QCT
NO TPNAME(S) ASSOCIATED WITH LUNAME
*90332/114253*
```

Entry ET:

```
/DISPLAY LUNAME 'network1.luname1' luname3 'nid2.luname4' OUTPUT
```

Response ET:

```
LUNAME                        ENQCT       DEQCT      QCT      CONVCT
NETWORK1.LUNAME1                  5           3        2           2
LUNAME3                           5           4        1           1
NID2.LUNAME4                      0           0        0           0
*92232/114253*
```

Entry ET:

```
  /DISPLAY LUNAME 'network1.luname1' TPNAME tpname1 badtpnme
```

Response ET:

```
LUNAME/TPNAME                 ENQCT       DEQCT      QCT
NETWORK1.LUNAME1
-TPNAME1                          0           0        0 STO
-BADTPNME           IS INVALID
*92232/114253*
```

Entry ET:

```
/DISPLAY LUNAME ALL INPUT
```

Response ET

```
LUNAME           #APPC-CONV
NET1.LU1                  0 STO,TRA
NET2.LU1                  0
LU3                       0 STO,TRA
NET4.LU3                  0 STO
NET5.LU5                  0 TRA
FUTURE INPUT:  STO,TRA
*93069/114435*
```

Entry ET:

```
/DISPLAY LUNAME ALL OUTPUT
```

Response ET:

```
LUNAME                ENQCT      DEQCT        QCT     CONVCT
NET1.LU1                  0          0          0          0  STO
NET2.LU1                  0          0          0          0  TRA
LU3                       0          0          0          0  STO,TRA
NET4.LU3                  0          0          0          0  TRA
NET5.LU5                  0          0          0          0  STO
FUTURE OUTPUT:  STO,TRA
*93069/114435*
```

Entry ET:

```
  /DISPLAY LUNAME LUNAMEA TPNAME TPNAME1 TPNAME2 QCNT
```

Response ET:

```
  LUNAME/TPNAME          GBLQCT
  LUNAMEA
  -TPNAME1                   25
  -TPNAME2           IS INVALID
  *95200/170817*
```

Entry ET:

```
  /DISPLAY LUNAME LUNAMEA1 LUNAMEB LUNAMEC QCNT
```

Response ET:

```
LUNAME       GBLQCT
LUNAMEA1 1000000000
LUNAMEB          25
LUNAMEC IS INVALID
*95200/170817*
```

Entry ET:

```
/DISPLAY LUNAME LUNAMEB TPNAME ALL QCNT
```

Response ET:

```
LUNAME/TPNAME          GBLQCT
LUNAMEB
-TPNAMEB1                  10
-TPNAMB1234567890+          0
 12345678901234567890123456789012345678901234567890
*95200/170817*
```

Example for /DISPLAY MASTER Command

Entry ET:

```
/DISPLAY MASTER
```

Response ET:

```
LTERM MASTER
PTERM    3-    1
*89117/130245*
```

Examples for /DISPLAY MODIFY Command

The following are examples of the /DISPLAY MODIFY command.

Example 1 for /DISPLAY MODIFY Command

In the following examples for /DISPLAY MODIFY, /MODIFY PREPARE has already succeeded.

Entry ET:

```
/DISPLAY MODIFY ALL
```

Response ET:

```
LIBRARY  IMSACBA  (A)  OLC.ACBLIB.ALL
LIBRARY  FORMATA  (I)  IMSQA.FMT1
LIBRARY  MODBLKSA (A)  I41RTS42.CMODBLKS
LIBRARY  MATRIXA  (A)  OLC1.MATRIX1
LIBRARY  IMSACBB  (I)  OLC.ACBLIB.ALL
LIBRARY  FORMATB  (A)  IMSQA.FMT1
LIBRARY  MODBLKSB (I)  I41RTS42.CMODBLK1
LIBRARY  MATRIXB  (I)  OLC1.MATRIX2
DATABASE OLCDB088      /DBR ACTIVE
DATABASE OLCDB101      PSB SCHEDULED
PROGRAM  OLCPB021      SCHEDULED
PROGRAM  OLCPB109      SCHEDULED
RTCODE   OLCRC056      ACTIVE
TRAN     OLCTB105      QUEUING     1
TRAN     CDEBTRN5      CONVERSATION TERM/USER    4-   2         ID= 0009
TRAN     OLCTB109      SCHEDULED
TRAN     CDEBTRN8      CONVERSATION TERM/USER L3270D            ID= 0002
TRAN     OLCTB111      QUEUING     5
TRAN     CDEBTRN2      CONVERSATION TERM/USER DYNT0001 IMSUS01  ID= 0005
TRAN     CDEBTRN3      CONVERSATION TERM/USER   12-   1         ID= 0008
TRAN     CDEBTRN1      CONVERSATION TERM/USER          IMSUS12  ID= 0001
DISPLAY MODIFY COMPLETE
*93336/093025*
```

Entry ET:

```
/DISPLAY MODIFY DBS
```

Response ET:

```
  DATABASE CALENDER      AREAS OPEN:
      AREA FEBRUARY
      AREA APRIL
      AREA JULY
      AREA AUGUST
      AREA SEPTEMBR
      AREA OCTOBER
      AREA DECEMBER
  DATABASE CALENDER      RANDOMIZER: RMOD3
  DEDBS    SHARING       RANDOMIZER: RMOD3
  DATABASE DEDBJN21      RANDOMIZER: RMOD3
  RMOD3    NOT           LOADED
  DATABASE DEDBJN22      RANDOMIZER: RMOD5
  RMOD5    NOT           LOADED
  DISPLAY MODIFY COMPLETE *95299/161529*
```

Entry ET:

```
/DISPLAY MODIFY TRS
```

Response ET:

```
  TRAN     OLCFT112      QUEUING     2
  TRAN     OLCFT115      QUEUING    23
  PROGRAM  OLCFP115      QUEUING     1
  TRAN     CDEBTRN8      GLOBAL QUEUE COUNT INTERNAL ERROR
  TRAN     OLCFT116      QUEUING     6
  DISPLAY MODIFY COMPLETE *96193/135935*
```

Explanation: Work is in progress that will cause online change to fail, because several transactions and a FP program have a global queue count on the shared queues. The global queue count for transaction CDEBTRN8 could not be determined because of an internal error.

Example 2 for /DISPLAY MODIFY Command

Entry ET:

```
/DISPLAY MODIFY MODS
```

Response ET:

```
DFS000I     FORMAT   0C01_FF02A _O ADDED
DFS000I     FORMAT   0C01_FF02B _O ADDED
DFS000I     FORMAT   0C7F_FN01A _O CHANGED
DFS000I     FORMAT   0C7F_FN01B _O CHANGED
DFS000I     FORMAT   0C7F_FN01C _O CHANGED
DFS000I     FORMAT   0C7F_FF04A _O DELETED
DFS000I     FORMAT   MF04E         DELETED
DFS000I     DATABASE OLCDB101      ADDED
DFS000I     DATABASE OLCDB102      ADDED
DFS000I     DATABASE OLCDB103      ADDED
DFS000I     DATABASE OLCDB104      ADDED
DFS000I     DATABASE OLCDB157      ADDED
DFS000I     DATABASE OLCDB111      DELETED
DFS000I     DATABASE OLCDB159      DELETED
DFS000I     DATABASE OLCDI105      DELETED
DFS000I     DATABASE OLCDI106      DELETED
DFS000I     DMB      OLCDB101      ADDED
DFS000I     DMB      OLCDB102      ADDED
DFS000I     DMB      DEDBJN24      CHANGED
DFS000I       AREA   DB24A000      CHANGED
DFS000I     DMB      OLCDB105      DELETED
```

```
DFS000I     PROGRAM  OLCFP107      ADDED
DFS000I     PROGRAM  OLCFP108      ADDED
DFS000I     PROGRAM  OLCPB101      ADDED
DFS000I     PROGRAM  OLCFP109      CHANGED
DFS000I     PROGRAM  OLCFP110      CHANGED
DFS000I     PROGRAM  OLCFP111      CHANGED
DFS000I     PROGRAM  OLCPB105      DELETED
DFS000I     PROGRAM  OLCPB106      DELETED
DFS000I     PSB      OLCFP107      ADDED
DFS000I     PSB      OLCFP108      ADDED
DFS000I     PSB      CDEBS         CHANGED
DFS000I     PSB      OLCPB105      DELETED
DFS000I     PSB      OLCPB106      DELETED
DFS000I     RT CODE  OLCFT108      ADDED
DFS000I     RT CODE  OLCFR119      CHANGED
DFS000I     RT CODE  OLCFR117      DELETED
DFS000I     TRAN     OLCFT108      ADDED
DFS000I     TRAN     OLCTBG17      ADDED
DFS000I     TRAN     CDEBTRNA      CHANGED
DFS000I     TRAN     CDEBTRN1      CHANGED
DFS000I     TRAN     CONV31X       DELETED
DFS000I     DISPLAY MODIFY COMPLETE *98356/094452*   SYS3
```

Explanation: Displays resources to be modified (added, changed, or deleted) by online change.

Example 3 for /DISPLAY MODIFY Command

Entry ET:

```
/DISPLAY MODIFY ALL
```

Response ET:

```
LIBRARY  IMSACBA  (A)   IMSTESTG.DELTALIB
                        (A)  IMSTESTL.TNUC2
                        (A)  IMSTESTG.IMS61RC.ACBLIB
                        (A)  IMSTESTG.IMS61R.ACBLIB
       LIBRARY  FORMATA  (A)  IMSTESTG.MFS.OVERRIDE.FORM
                        (A)  IMSTESTG.MFS.FORMAT
                        (A)  IMSQA.FMT1
       LIBRARY  MODBLKSA (A)  IMSBLD.I61RTS25.CMODBLKS
       LIBRARY  MATRIXA  (A)  IMSTESTG.I61RTS25.MATRIX
       LIBRARY  IMSACBB  (I)  IMSTESTG.DELTALIB
                        (I)  IMSTESTL.TNUC3
                        (I)  IMSTESTG.IMS61RC.ACBLIB
                        (I)  IMSTESTG.IMS61R.ACBLIB
       LIBRARY  FORMATB  (I)  IMSTESTG.MFS.OVERRIDE.FORM
                        (I)  IMSTESTG.MFS.FORMAT
                        (I)  IMSQA.FMT1
       LIBRARY  MODBLKSB (I)  IMSBLD.I61RTS25.CMODBLKS
       LIBRARY  MATRIXB  (I)  IMSTESTG.I61RTS25.MATRIX
       DATABASE DEDBJN21      RANDOMIZER: RMOD3        (1)
       RMOD3    LOADED                                 (2)
       DEDBS    SHARING       RANDOMIZER: RMOD3        (3)
           DEDB DEDBJN22
           DEDB DEDBJN23
           DEDB DEDB3301
       DATABASE DEDBJN22      RANDOMIZER: RMOD3
       RMOD3    NOT           LOADED
       DEDBS    SHARING       RANDOMIZER: RMOD3
           DEDB DEDBJN21
           DEDB DEDBJN23
           DEDB DEDB3301
       DATABASE DEDBJN23      RANDOMIZER: RMOD3
       RMOD3    LOADED
       DEDBS    SHARING       RANDOMIZER: RMOD3
           DEDB DEDBJN21
```

```
          DEDB DEDBJN22
          DEDB DEDB3301
      NO WORK PENDING *99328/110535*   SYS3
```

Explanation: In the previous example:

- DEDB databases DEDBJN21, DEDBJN22, and DEDBJN23 have undergone Online Change.
- After a database line for DEDBS displays, a line follows containing randomizer status (loaded or not loaded).
- After the randomizer status displays, a line might follow containing `DEDBS SHARING RANDOMIZER`; this line is followed by DEDB lines listing the DEDBs that share the randomizer.

Entry ET:

```
/DISPLAY MODIFY ALL
```

Response ET:

```
LIBRARY  IMSACBA   (A)  IMSTESTG.DELTALIB
                  (A)  IMSTESTL.TEMPA.ACBLIB
                  (A)  IMSTESTG.IMS800C.ACBLIB
                  (A)  IMSTESTG.IMS800.ACBLIB
                  (A)  IMSTESTG.MFS.OVERRIDE.FORMAT
LIBRARY  FORMATA  (A)  IMSTESTG.MFS.FORMAT
                  (A)  IMSQA.FMT1
LIBRARY  MODBLKSA (A)  TEMPA.MODBLKS
LIBRARY  MATRIXA  (A)  IMSTESTG.I800TS2A.MATRIX
LIBRARY  IMSACBB  (I)  IMSTESTG.DELTALIB
                  (I)  IMSTESTL.TEMPB.ACBLIB
                  (I)  IMSTESTG.IMS800C.ACBLIB
                  (I)  IMSTESTG.IMS800.ACBLIB
                  (I)  IMSTESTG.MFS.OVERRIDE.FORMAT
LIBRARY  FORMATB  (I)  IMSTESTG.MFS.FORMAT
                  (I)  IMSQA.FMT1
LIBRARY  MODBLKSB (I)  TEMPB.MODBLKS
LIBRARY  MATRIXB  (I)  IMSTESTG.I800TS2A.MATRIX
DATABASE DEDBJN21      NOT OPEN, BUT DBR NEEDED ON TRACKER  (1)
DATABASE DEDBJN21      RANDOMIZER: RMOD3
RMOD3    NOT           LOADED
DATABASE DEDBJN22      RANDOMIZER: RMOD5
RMOD5    NOT           LOADED
DATABASE DEDBJN24      RANDOMIZER: RMOD4
RMOD4    NOT           LOADED
DATABASE THFP1WEB      RANDOMIZER: DBFHDC44
DBFHDC44 NOT           LOADED
DISPLAY       MODIFY COMPLETE *01303/085213*   SYS3
```

Explanation: The DEDB database DEDBJN21 is prepared to be online changed on a tracker environment. If the AREA is not stopped and the database is not open, the AREA can be open if log records type 5701 or 5950 are routed to the RSR tracking IMS from the active IMS while Online Change is in progress. It is then required that database be DBRed in the case of database level change. The tracking suspend point (Suspended Log Sequence Number) is registered in RECON when the AREA is stopped. After the online change committed sucessfully, and when the /STA AREA command is entered, the Online Forward Recovery will open the AREA and all log records from the suspended point will be applied to the AREA.

Example for /DISPLAY MSNAME Command

Entry ET:

```
/DISPLAY MSNAME ALL
```

Response ET:

```
  MSNAME    ENQCT   DEQCT    QCT
  SYSTEM2A      0       0       0
  SYSTEM2B      0       0       0
  FRESNO        0       0       0
  BOSTON        0       0       0
  ALBANY        0       0       0
  TEST1         0       0       0
  *91350/053859*
```

Entry ET:

```
  /DISPLAY MSNAME ALL
```

Response ET:

```
MSNAME    ENQCT   DEQCT    QCT   IMS1
LINK23B4      0       0       0  DYNAMIC   IMS1
LINK12V1      0       0       0    IMS1
LINK23U2      0       0       0  DYNAMIC   IMS1
ELINK211      0       0       0  DYNAMIC   IMS1
LINK13B3      0       0       0    IMS1
LINK13V       0       0       0    IMS1
LINK12V       0       0       0    IMS1
LINK13C2      0       0       0    IMS1
LINK13X       0       0       0    IMS1
LINK12V2      0       0       0    IMS1
LINK13M2      0       0       0    IMS1
LINK13V2      0       0       0    IMS1
ELINK124      0       0       0    IMS1
LINK23V1      0       0       0  DYNAMIC   IMS1
LINK12B1      0       0       0    IMS1
LINK12B2      0       0       0    IMS1
LINK12Y       0       0       0    IMS1
ELINK120      0       0       0    IMS1
LINK13B4      0       0       0    IMS1
LINK21V3      0       0       0  DYNAMIC   IMS1
ELINK1MS      0       0       0    IMS1
LINK12M1      0       0       0    IMS1
LINK23V       0       0       0  DYNAMIC   IMS1
LINK13U2      0       0       0    IMS1
LINK23B3      0       0       0  DYNAMIC   IMS1
ELINK224      0       0       0  DYNAMIC   IMS1
LINK23X       0       0       0  DYNAMIC   IMS1
LINK23V2      0       0       0  DYNAMIC   IMS1
ELINK121      0       0       0    IMS1
LINK23C2      0       0       0  DYNAMIC   IMS1
LINK12U1      0       0       0    IMS1
LINK23M2      0       0       0  DYNAMIC   IMS1
ELINK122      0       0       0    IMS1
ELINK222      0       0       0  DYNAMIC   IMS1
LINK12C1      0       0       0    IMS1
LINK13V1      0       0       0    IMS1
ELINK123      0       0       0    IMS1
PTH3TSTB      0       0       0    IMS1
*97258/175636*
```

Examples for /DISPLAY NODE Command

The following are examples of the /DISPLAY NODE command.

Example 1 for /DISPLAY NODE Command

Entry ET:

```
  /DISPLAY NODE DTSLU* LUP1 ENDS01 LU6NDPA LU6NDPH
```

Response ET:

```
NODE-USR TYPE   CID       RECD ENQCT DEQCT   QCT  SENT   SYS3
DTSLU002 SLUP   010000D3     0     0     0     0     0 SIGN(IMSUS05 )
                                                       IDLE CON
DTSLU202 SLU2   010000CB     0     0     0     0     1 SIGN(IMSUS02 )
                                                       IDLE CON
DTSLU603 LUT6
-IMSUS04        010000C7     0     0     0     0     0 SIGN(IMSUS04 )
                                                       IDLE CON PRI
DTSLU205 SLU2   010000A5     2     1     1     0     3 SIGN(IMSUS14 )
                                                       IDLE CONV-ACT CON
DTSLU204 SLU2   010000A4     3     0     0     0     4 SIGN(IMSUS13 )
                                                       IDLE CON EXC
DTSLU203 SLU2   010000A3     3     0     0     0     4 SIGN(IMSUS12 )
                                                       PRST(WTOR    )
                                                       IDLE MFST CON
LUP1     SLUP   010000CD     0     0     0     0     0 IDLE CON STATIC
ENDS01   SLU2   010000A6     1     0     0     0     2 SIGN(IMSUS08 )
                                                       IDLE CON STATIC
LU6NDPA  LUT6
-LU6SPG         010000BB     0     0     0     0     0 IDLE CON PRI
-CA12           010000BD     0     0     0     0     0 IDLE CON PRI
-CA11           010000BF     0     0     0     0     0 IDLE CON PRI
LU6NDPH  LUT6
-CA14           010000C1     0     0     0     0     0 IDLE CON PRI
-N/A            00000000     0     0     0     0     0 IDLE STATIC
-N/A            00000000     0     0     0     0     0 IDLE STATIC
*90240/134730*
```

Explanation: User IMSUS05 is signed on to dynamic node DTSLU002. User IMSUS02 is signed on to dynamic node DTSLU202. User IMSUS04 is allocated to ISC node DTSLU603. User IMSUS14 is signed on to dynamic node DTSLU205 and has an active conversation. User IMSUS13 is signed on to dynamic node DTSLU204 and is in exclusive mode. User IMSUS12 is signed on to dynamic node DTSLU203 and has a preset destination of LTERM WTOR and is in MFS test mode. Static node LUP1 is logged on. User IMSUS08 is signed on to static node ENDS01. Users LU6SPG, CA12, and CA11 are allocated to static ISC node LU6NDPA, which means node LU6NDPA has 3 active parallel sessions. User CA14 is allocated to static ISC node LU6NDPH, which means node LU6NDPH has one active parallel session.

Example 2 for /DISPLAY NODE Command

Entry ET:

```
/DISPLAY NODE NSLUTP1 NSLUTP6 NPLUTP6
```

Response ET:

```
NODE-USR TYPE    CID        RECD ENQCT DEQCT   QCT  SENT
NSLUTP1  SLU1    00000000      0     0     0     0     0 IDLE
NSLUTP6  LUT6    00000000      5     3     2     1     5 IDLE
NPLUTP6  LUT6
-A               00000000      1     1     1     0     1 IDLE FORCE
-B               00000000     10    25    13    12     9 IDLE
-N/A             00000000      0     0     0     0     0 IDLE
-C               00000000      0     4     4     0     4 IDLE
*90229/083059*
```

Entry ET:

```
/DISPLAY NODE ALL
```

Response ET:

```
  NODE-USR TYPE   CID       RECD ENQCT DEQCT   QCT  SENT   SYS3
  WRIGHT   SLU2   00000000     0     0     0     0     0 IDLE
  LUNS01   NTO    00000000     0     0     0     0     0 IDLE
```

```
WS12     FIN    00000000     0     0     0     0     0 IDLE
VR3275A1 3277   00000000     0     0     0     0     0 IDLE
VR3286A1 3286   00000000     0     0     0     0     0 IDLE
LUP1     SLUP   00000000     0     0     0     0     0 IDLE
Q3290A   SLU2   00000000     0     0     0     0     0 IDLE
LU6FP02  LUT6   00000000     0     0     0     0     0 IDLE
LU6NDPA  LUT6
-N/A             00000000     0     0     0     0     0 IDLE
-N/A             00000000     0     0     0     0     0 IDLE
-N/A             00000000     0     0     0     0     0 IDLE
NDNTOXA1 NTO    00000000     0     0     0     0     0 ASR
ND3287A1 SLU1   00000000     0     0     0     0     0 ASR
ND3287A2 SLU1   00000000     0     0     0     0     0 IDLE
ND3287A3 SLU1   00000000     0     0     0     0     0 IDLE
ND3287A4 SLU1   00000000     0     0     0     0     0 ASR
NDSLU2A1 SLU2   00000000     0     0     0     0     0 ASR
NDSLU2A2 SLU2   00000000     1     1     0     1     1 ASR
NDSLUPA1 SLUP   00000000     0     0     0     0     0 ASR
NDSLUPA2 SLUP   00000000     0     0     0     0     0 IDLE
NDLU6XSG LUT6   00000000     0     0     0     0     0 IDLE
NDLU6XPG LUT6
-N/A             00000000     0     0     0     0     0 ASR
-N/A             00000000     0     0     0     0     0 ASR
-N/A             00000000     0     0     0     0     0 ASR
-N/A             00000000     0     0     0     0     0 ASR
*91213/100107*   SYS3
```

Entry ET:

```
/DISPLAY NODE IN902D1 IN902D4
```

Response ET:

```
NODE-USR TYPE    CID        RECD ENQCT   QCT  SENT
IN902D1  SLU1    100121C5    125   125     0   125 CON
*91012/192544*
```

Entry ET:

```
/DISPLAY NODE NLUTP6P USER C
```

Response ET:

```
NODE-USR TYPE    CID        RECD ENQCT DEQCT   QCT  SENT
NLUTP6P  LUT6
-C               00000000      0    10     8     2     8
*90229/083059*
```

Example 3 for /DISPLAY NODE Command

The following commands illustrate the use of /DISPLAY to monitor changes in the mode table values for a node.

Entry ET:

```
/DISPLAY NODE LUTYPEP1 MODE
```

Response ET:

```
NODE-USR TYPE   DEF MODETBL  ACT MODETBL
LUTYPEP1 SLUP   DEFRESP
*90179/100206*
```

Explanation: DEFRESP is the mode table name defined for node LUTYPEP1 at system definition. The session is not active so the ACT MODETBL field is blank.

Entry ET:

```
/OPN NODE LUTYPEP1
```

Response ET:

```
DFS058I OPNDST COMMAND COMPLETED
```

Entry ET:

```
/DISPLAY NODE LUTYPEP1 MODE.
```

Response ET:

```
NODE-USR TYPE   DEF MODETBL  ACT MODETBL
LUTYPEP1 SLUP   DEFRESP      DEFRESP
*90179/100508*
```

Explanation: A mode table name was not specified with the /OPNDST command so the default value defined at system definition was used to initiate the session.

Entry ET:

```
/CLS NODE LUTYPEP1
```

Response ET:

```
DFS058I CLSDST COMMAND COMPLETED
```

Entry ET:

```
/DISPLAY NODE LUTYPEP1 MODE
```

Response ET:

```
NODE-USR TYPE   DEF MODETBL  ACT MODETBL
LUTYPEP1 SLUP   DEFRESP
*90179/100630*
```

Explanation: Active mode table name displays as blank at normal session termination.

Entry ET:

```
/OPN NODE LUTYPEP1 MODE ALPHA.
```

Response ET:

```
DFS058I OPNDST COMMAND COMPLETED
```

Entry ET:

```
/DISPLAY NODE LUTYPE1 MODE
```

Response ET:

```
NODE-USR TYPE   DEF MODETBL  ACT MODETBL
LUTYPEP1 SLUP  DEFRESP       ALPHA
*90179/100805*
```

Explanation: The mode table name specified with the /OPNDST command (ALPHA) is used to initiate the session. The default value specified at system definition (DEFRESP) is overridden by the /OPNDST command.

Example 4 for /DISPLAY NODE Command

The following example shows the display of all nodes with the automatic session restart designation.

Entry ET:

```
/DISPLAY NODE AUTOSR
```

Response ET:

```
NODE-USR TYPE    CID       RECD ENQCT DEQCT   QCT  SENT
WS12     FIN     00000000     0     0     0     0     0 ASR
NODE0005 FIN     00000000     0     0     0     0     0 ASR
NODE0006 FIN     00000000     0     0     0     0     0 ASR
NODE0007 FIN     00000000     0     0     0     0     0 ASR
NTOLUNS  NTO     00000000     0     0     0     0     0 ASR IDLE
NTO2741  NTO     00000000     0     0     0     0     0 ASR IDLE
LU6NODE2 LUT6
-N/A             00000000     0     0     0     0     0 ASR IDLE
-N/A             00000000     0     0     0     0     0 ASR IDLE
-N/A             00000000     0     0     0     0     0 ASR IDLE
*90179/101609*
```

Example 5 for /DISPLAY NODE Command

Entry ET:

```
/DISPLAY NODE NODE1 NODE2 QCNT
```

Response ET:

```
NODE         GBLQCT
NODE1        100000
NODE2            25
*95200/170817*
```

Entry ET:

```
/DISPLAY NODE NODE1 USER USER1 USER2  QCNT
```

Response ET:

```
NODE/USER     GBLQCT
NODE1
-USER1            10
-USER2             0
*95200/170817*
```

Example 6 for /DISPLAY NODE Command

Entry ET:

```
/DISPLAY NODE FPEN0001 EMHQ QCNT
```

Response ET:

```
LTERM        GBLQCT
FPEN0001          0
*98203/143542*
```

Example 7 for /DISPLAY NODE RECOVERY Command

Entry ET:

```
/DISPLAY NODE DTSLU* DTSLU202 L3270A RECOVERY
```

Response ET:

```
NODE-USR OWNER SRM    CONV STSN FPATH
DTSLU002 IMSA  GLOBAL Y    Y    Y
DTSLU603
-IMSUS04 IMSA LOCAL   Y    Y    Y
DTSLU202 IMSB GLOBAL  Y    N    N
L3270A   IMSA LOCAL   Y    N    Y
*99240/134730*
```

Example for /DISPLAY OASN Command

Entry ET:

```
/DISPLAY OASN SUBSYS ALL
```

Response ET:

```
SUBSYS   OASN
SSTR
         0000000001
*90207/095627*
```

Example for /DISPLAY OLDS Command

Entry ET:

```
/DISPLAY OLDS
```

Response ET:

```
OLDS-DDNAME % FULL RATE ARCH-JOB    ARCH-STATUS  OTHER-STATUS
*DFSOLP00       37    9                          IN USE
 DFSOLP02                          AVAILABLE
 DFSOLP01                          AVAILABLE
 SINGLE OLDS LOGGING, SINGLE WADS LOGGING
 AUTOMATIC ARCHIVE = 01
 WADS = *DFSWADS0 DFSWADS1
 SLDSREAD ON
 *89184/141407*
```

Examples for /DISPLAY OTMA Command

The following are examples of the /DISPLAY OTMA command.

Example 1 for /DISPLAY OTMA Command

Entry ET:

```
/DISPLAY OTMA
```

Response ET:

```
GROUP/MEMBER      XCF-STATUS   USER-STATUS    SECURITY
HOANG
-APPL8            ACTIVE       SERVER         FULL
-CLIENT1          ACTIVE       ACCEPT TRAFFIC
*94165/165753*
```

Example 2 for /DISPLAY OTMA Command

Entry ET:

```
/DISPLAY OTMA
```

Response ET:

```
GROUP/MEMBER      XCF-STATUS   USER-STATUS    SECURITY
HOANG
-APPL8            NOT DEFINED  SERVER         FULL
-CLIENT1          ACTIVE       ACCEPT TRAFFIC
*94165/165753*
```

Example for /DISPLAY OVERFLOWQ Command

Entry ET:

```
/DISPLAY OVERFLOWQ STRUCTURE ALL
```

Response ET:

```
STRUC-RSCTYPE    OFLSTRUC-RSCNAME LUNAME-TMEMBER    TPNAME-TPIPE
IMSMSGQ01        IMSMSGQ01OFLW
 TRANSACTION      TRANA
 LTERM            LTERMX
 SUSPENDTRAN      TRANA
 APPC             TMSTMP- IMSA    LUNAME1           TPNAME1
 SERIALTRAN       TRANABC IMSA
 APPC             TMSTMP- IMSA
 OTMA             TMSTMP- IMSA    TMEMBERA          TPIPEA
 APPC             TMSTMP- IMSA    NETWORK2.LUNAME2  TPNM123456789012+
                                                    3456789012345678+
                                                    901234567890
 OTMA             TMSTMP- IMSB
 REMOTE           TRANB
 REMOTE           LTERMA
 UNKNOWN          ASIXTEENCHARNAME
 REMOTE           MSNAMEA
 IMSEMHQ01        OVERFLOW STRUCTURE IS NOT DEFINED
  BALGRP           FPPSB1
  LTERM            LTERMY
  BALGRP           FPPSB2
  *95200/170817*
```

Examples for /DISPLAY POOL Command

The following are examples of the /DISPLAY POOL command.

Example 1 for /DISPLAY POOL Command

Entry ET:

```
/DISPLAY POOL ALL
```

Response ET:

```
NAME SP#     CURR      MAX     GETS    FREES
IOSB 228      28K      28K        3        0
GIOB 228       0K       0K        0        0
OSWA 228       8K       8K        1        0
GOWA 228       0K       0K        0        0
PST  231      52K      56K       14        1
DPST 231      40K      44K       11        1
SAP  231     132K     132K       32        0
GQMW 231      16K      16K        4        0
LQMW   0     324K     324K       81        0
D1WA 228       8K       8K        2        0
DL2W   0       8K       8K        2        0
DG2W 231       4K       4K        1        0
QSAV 231       4K      16K        3        2
VRPL 231      16K      16K        4        0
LSAV   0     808K     808K      201        1
AWE  231     112K     112K       28        0
PDIR 231      36K      36K        1        0
DDIR 231      28K      28K        1        0
LCRE 241       4K       4K        1        0
PCIB   0       0K       0K        0        0
SIDX 241       4K       4K        1        0
RRE  241       4K       4K        1        0
SMB  231      62K      62K        1        0
BCPT 231       4K       4K        1        0
GESE 241       0K       0K        0        0
EPST 231     208K     208K       20        0
IDT  241       0K       0K        0        0
DBRC 231       0K       0K        0        0
TTAB 231     224K     224K       56        0
EZS  241       0K       0K        0        0
FNCB 231       0K       0K        0        0
```

```
TCBT 231       4K       4K        1        0
CMWU 231      12K      12K        3        0
FSRB 228       4K       4K        1        0
KLSD   0       0K       0K        0        0
IRLM 231       0K       0K        0        0
STTR 231      52K      52K       13        0
BQEL 231       0K       0K        0        0
USMU 229       0K       0K        0        0
RCTE 231       3K       3K        1        0
SLOG 231       4K       4K        1        0
ADSC 228       0K       0K        0        0
FPCP 231       0K       0K        0        0
AESL 231       0K       0K        0        0
XPST 231      16K      16K        4        0
BXQE 231      16K      16K        4        0
SRBC 228       8K       8K        2        0
MSGP 241       4K       4K        1        0
LPST 251       8K       8K        2        0
CULE 231       4K       4K        1        0
RPST 241       4K       4K        1        0
VTCB 251      18K      18K        0        0
RECA   0      19K      19K        1        0
GSAV 231       0K       0K        0        0
FEIB 231       0K       0K        0        0
IEQE   0       0K       0K        0        0
L56X 231       4K       4K        1        0
X124 241       4K       4K        1        0
STAT 241       4K       4K        1        0
EQEL 241       0K       0K        0        0
TT24 231      24K      24K        6        0
LGWA 231     104K     104K       26        0
VWA  231       8K       8K        2        0
LGWX   0       0K       0K        0        0
LQB  251     104K     104K        0        0
RCNT 251       2K       2K        0        0
DDRE   0       0K       0K        0        0
CCB    0       0K       0K        0        0
LGND 251       8K       8K        2        0
USRD 251       4K       4K        1        0
LS24   0       2K       2K        1        0
GS24 231       0K       0K        0        0
CLLE 231      64K      64K       16        1
QMBA   0       8K       8K        1        0
AHDR 251       0K       0K        0        0
XMCI 241     180K     180K       45        1
DBPB 231       0K       0K        0        0
LG24   0      12K      12K        3        0
IAFP 231       0K       0K        0        0
RACW 231       0K       0K        0        0
LUB  251       0K       0K        0        0
TIB  251      28K      28K        1        0
DESC 251       0K       0K        0        0
PF62 251       0K       0K        0        0
QAB  251       0K       0K        0        0
LCLL   0     436K     436K      109        0
CBLK 251       0K       0K        0        0
SVPG 231      24K      24K        6        0
SVPL   0     148K     148K       37        0
SOPB   0       0K       0K        0        0
CBT POOLS        GLBL    1541K  LCL    2236K
MESSAGE QUEUE POOL:   BFRS/SIZE 20/2112
 ENQ      6  DEQ      6  CAN     30  WAIT      0  I/O     15   ERR   0
 QBLKS:  MAX # RECORDS AVAIL     1332  CUR IN USE        3 =  0 %
 SMSGQ:  MAX # RECORDS AVAIL     5940  CUR IN USE        4 =  0 %
 LMSGQ:  MAX # RECORDS AVAIL     1350  CUR IN USE        4 =  0 %
MESSAGE FORMAT POOL:  SIZE     49152  SPACE     48576  DIRS     9011
  REQ1          2  I/O           2  DIR          4
```

```
  WAIT          2  FREE      48432  ERR          0
SEQUENTIAL BUFFERING:  STATUS = NOT INIT
  MAX    N.A.  FREE   N.A.  CURR      0K HIGH      0K
OSAM DB BUFFER POOL: ID O04K   BSIZE   4K NBUF  1000 FX=Y/Y
  LCTREQ   1765296     NEWBLK         0     ALTREQ        340800
  PURGRQ     39371     FNDIPL   1370897     BFSRCH       1987604
  RDREQ     378355     BFSTLW         0     PURGWR        150284
  WBSYID      1431     WBSYWR         0     WBSYRD             0
  WRLSEO       296     WNOBFR         0     ERRORS   00000/00000
OSAM DB BUFFER POOL: ID O08K   BSIZE   8K NBUF  100 FX=Y/Y
  LCTREQ    228080     NEWBLK         0     ALTREQ             0
  PURGRQ         0     FNDIPL    204190     BFSRCH        316566
  RDREQ      23891     BFSTLW         0     PURGWR             0
  WBSYID         0     WBSYWR         0     WBSYRD             0
  WRLSEO         0     WNOBFR         0     ERRORS   00000/00000
OSAM DB BUFFER POOL: ID O12K   BSIZE   12K NBUF  100 FX=Y/Y
  LCTREQ     83282     NEWBLK         0     ALTREQ          6896
  PURGRQ      4384     FNDIPL     70743     BFSRCH         81395
  RDREQ       7622     BFSTLW         0     PURGWR          6118
  WBSYID         0     WBSYWR         0     WBSYRD             0
  WRLSEO         0     WNOBFR         0     ERRORS   00000/00000
OSAM DB BUFFER POOL: BSIZE ALL NBUF  1200 OSM=     6000K
  LCTREQ   2076667     NEWBLK         0     ALTREQ        347697
  PURGRQ     43755     FNDIPL   1645837     BFSRCH       2385576
  RDREQ     409869     BFSTLW         0     PURGWR        156402
  WBSYID      1431     WBSYWR         0     WBSYRD             0
  WRLSEO       296     WNOBFR         0     ERRORS   00000/00000
VSAM DB BUFFER POOL: ID VLP1  BSIZE   2K TYPE D  FX=N/Y/N
  RRBA           0     RKEY           0     BFALT              0
  NREC           0     SYNC PT    29446     NBUFS            500
  VRDS        1253     FOUND          0     VWTS              68
  HSR-S          0     HSW-S          0     HS NBUFS           0
  HS R/W-FAIL  000000/000000   ERRORS 000000/000000
VSAM DB BUFFER POOL: ID VLP1 BSIZE    4K TYPE D  FX=N/Y/N
  RRBA         370     RKEY      187583     BFALT              0
  NREC       10750     SYNC PT    29446     NBUFS           1000
  VRDS      145632     FOUND          0     VWTS            9771
  HSR-S          0     HSW-S         50     HS NBUFS          50
  HS R/W-FAIL  000000/000000   ERRORS 000000/000000
VSAM DB BUFFER POOL: ID VPL1 BSIZE    8K TYPE D  FX=N/Y/N
  RRBA        7375     RKEY        6704     BFALT              0
  NREC           0     SYNC PT    29446     NBUFS            100
  VRDS        7362     FOUND          0     VWTS               0
  HSR-S          0     HSW-S          0     HS NBUFS           0
  HS R/W-FAIL  000000/000000   ERRORS 000000/000000
VSAM DB BUFFER POOL: BSIZE ALL  VS=    5800K HS=       200K
  RRBA        7745     RKEY      194287     BFALT              0
  NREC       10752     SYNC PT    29446     NBUFS           1600
  VRDS      154247     FOUND          0     VWTS            9839
  HSR-S          0     HSW-S         50     HS NBUFS          50
  HS R/W-FAIL  000000/000000   ERRORS 000000/000000
DMPB BUFFER POOL:
 SIZE     200K  FREE     200K  HIGH       0K
PSBP BUFFER POOL:
 SIZE      80K  FREE      80K  HIGH       0K
DPSB BUFFER POOL:
 SIZE     160K  FREE     160K  HIGH       0K
CIOP BUFFER POOL:
 SIZE      14K  HIGH      65K  LIMIT     NONE  OVERFLOW       0K
MAIN BUFFER POOL:
 SIZE      40K  FREE      39K  HIGH       1K
SPAP BUFFER POOL:
 SIZE       0K  HIGH       0K  LIMIT    9999K  OVERFLOW       0K
PSBW BUFFER POOL:
 SIZE     600K  FREE     600K  HIGH       0K
DBWP BUFFER POOL:
 SIZE      80K  FREE      80K  HIGH       0K
```

```
HIOP BUFFER POOL:
 SIZE     214K  HIGH     420K  LIMIT    9999K  OVERFLOW       0K
FPDB BUFFER POOL:
 AVAIL =    60  WRITING =     0  PGMUSE =     0  UNFIXED =    90
POOLNAME CISIZE PBUF  SBUF   MAX CURRENT LK HITS VALID
 1024  01024  00064 00016 00256 00064  N    NA   NA
 2048  02048  00064 00016 00256 00064  N    NA   NA
CESS BUFFER POOL:
 POOL IS NOT AVAILABLE FOR STATISTICS
EMHB BUFFER POOL:
 SIZE       0K  HIGH       0K  LIMIT    NONE   OVERFLOW       0K
FPWP BUFFER POOL:
 SIZE       0K  HIGH       0K  LIMIT    NONE   OVERFLOW       0K
EPCB BUFFER POOL:
 SIZE     300K  FREE     300K  HIGH       0K
LUMP BUFFER POOL:
 SIZE      68K  HIGH      68K  LIMIT    NONE   OVERFLOW       0K
LUMC BUFFER POOL:
 SIZE       0K  HIGH      32K  LIMIT    NONE   OVERFLOW       0K
*92120/134306*
```

Example 2 for /DISPLAY POOL Command

Entry ET:

```
/DISPLAY POOL AOIP
```

Response ET:

```
AOIP BUFFER POOL:
 SIZE      32K  HIGH      32K  LIMIT     NONE  OVERFLOW       0K
*92280/095545*
```

Example 3 for /DISPLAY POOL Command

Entry ET:

```
/DISPLAY POOL CBT
```

Response ET:

```
NAME SP#     CURR      MAX     GETS    FREES
IOSB 228      20K      20K        1        0
GIOB 228       0K       0K        0        0
OSWA 228       8K       8K        1        0
GOWA 228       0K       0K        0        0
PST  231      52K      56K       14        1
DPST 231      40K      44K       11        1
SAP  231      28K      28K        6        0
GQMW 231      12K      12K        3        0
LQMW   0      40K      40K       10        0
D1WA 228       8K       8K        2        0
DL2W   0       8K       8K        2        0
DG2W 231       4K       4K        1        0
QSAV 231       4K      12K        2        1
VRPL 231      16K      16K        4        0
LSAV   0      68K      72K       17        1
AWE  231      12K      12K        3        0
PDIR 231      36K      36K        1        0
DDIR 231      28K      28K        1        0
LCRE 241       4K       4K        1        0
PCIB   0       0K       0K        0        0
SIDX 241       4K       4K        1        0
RRE  241       4K       4K        1        0
SMB  231      61K      61K        1        0
BCPT 231       4K       4K        1        0
GESE 241       0K       0K        0        0
EPST 231     144K     144K       20        0
IDT  241       0K       0K        0        0
DBRC 231       0K       0K        0        0
```

```
TTAB 231     184K     184K       46        0
EZS  241       0K       0K        0        0
FNCB 231       0K       0K        0        0
TCBT 231       4K       4K        1        0
CMWU 231       8K       8K        2        0
FSRB 228       4K       4K        1        0
KLSD   0       0K       0K        0        0
IRLM 231       0K       0K        0        0
STTR 231      52K      52K       13        0
BQEL 231       0K       0K        0        0
USMU 229       0K       0K        0        0
RCTE 231       3K       3K        1        0
SLOG 231       4K       4K        1        0
ADSC 228       0K       0K        0        0
FPCP 231       0K       0K        0        0
AESL 231       0K       0K        0        0
XPST 231      16K      16K        4        0
BXQE 231      16K      16K        4        0
SRBC 228       8K       8K        2        0
MSGP 241       4K       4K        1        0
LPST 251       8K       8K        2        0
CULE 231     152K     152K        1        0
RPST 241       4K       4K        1        0
VTCB 251     305K     305K        0        0
RECA   0      19K      19K        1        0
GSAV 231       0K       0K        0        0
FEIB 231       0K       0K        0        0
IEQE   0       0K       0K        0        0
L56X 231       4K       4K        1        0
X124 241       4K       4K        1        0
STAT 241       4K       4K        1        0
EQEL 241       0K       0K        0        0
TT24 231      24K      24K        6        0
LGWA 231     104K     104K       26        0
VWA  231       8K       8K        2        0
LGWX   0       0K       0K        0        0
LQB  251     104K     104K        0        0
RCNT 251       4K       4K        0        0
DDRE   0       0K       0K        0        0
CCB    0       0K       0K        0        0
LGND 251       8K       8K        2        0
USRD 251       4K       4K        1        0
LS24   0       2K       2K        1        0
GS24 231       0K       0K        0        0
CLLE 231      60K      64K       16        1
QMBA   0       8K       8K        1        0
AHDR 251       0K       0K        0        0
XMCI 241      32K      36K        9        1
DBPB 231       0K       0K        0        0
LG24   0      12K      12K        3        0
IAFP 231       0K       0K        0        0
RACW 231       0K       0K        0        0
LUB  251       0K       0K        0        0
TIB  251      28K      28K        1        0
DESC 251       4K       4K        1        0
PF62 251       0K       0K        0        0
QAB  251       0K       0K        0        0
LCLL   0      36K      36K        9        0
CBLK 251       0K       0K        0        0
CBT POOLS        GLBL    1187K  LCL     658K
*91057/132506*
```

Example 4 for /DISPLAY POOL Command

Entry ET:

```
/DISPLAY POOL CIOP SPAP HIOP CESS EMHB FPWP LUMP LUMC
```

Response ET:

```
CIOP BUFFER POOL:
 SIZE      16K  HIGH      96K  LIMIT     999K  OVERFLOW      0K
SPAP BUFFER POOL:
 SIZE       0K  HIGH       0K  LIMIT    9999K  OVERFLOW      0K
HIOP BUFFER POOL:
 SIZE     128K  HIGH     128K  LIMIT    9999K  OVERFLOW      0K
CESS BUFFER POOL:
 POOL IS NOT AVAILABLE FOR STATISTICS
EMHB BUFFER POOL:
 SIZE       0K  HIGH       0K  LIMIT    9999K  OVERFLOW      0K
FPWP BUFFER POOL:
 SIZE       0K  HIGH       0K  LIMIT    9999K  OVERFLOW      0K
LUMP BUFFER POOL:
 SIZE      32K  HIGH       32K  LIMIT    NONE   OVERFLOW       OK
LUMC BUFFER POOL:
 SIZE      OK   HIGH       32K  LIMIT    NONE   OVERFLOW       OK
*91051/161116*
```

Example 5 for /DISPLAY POOL Command

Entry ET:

```
/DISPLAY POOL DBAS
```

Response ET:

```
SEQUENTIAL BUFFERING:   STATUS = NOT INIT
  MAX    N.A.  FREE   N.A.  CURR      0K HIGH      0K
OSAM DB BUFFER POOL:  ID   BSIZE   2K NBUF    4 FX=N/N
  LCTREQ      0        NEWBLK     0    ALTREQ    0
  PURGRQ      0        FNDIPL     0    BFSRCH    0
  RDREQ       0        BFSTLW     0    PURGWR    0
  WBSYID      0        WBSYWR     0    WBSYRD    0
  WRLSEO      0        WNOBFR     0    ERRORS 00000/00000
OSAM DB BUFFER POOL:  ID   BSIZE   6K NBUF    4 FX=N/N
  LCTREQ      0        NEWBLK     0    ALTREQ    0
  PURGRQ      0        FNDIPL     0    BFSRCH    0
  RDREQ       0        BFSTLW     0    PURGWR    0
  WBSYID      0        WBSYWR     0    WBSYRD    0
  WRLSEO      0        WNOBFR     0    ERRORS 00000/00000
OSAM DB BUFFER POOL:  ID   BSIZE   8K NBUF    4 FX=N/N
  LCTREQ      0        NEWBLK     0    ALTREQ    0
  PURGRQ      0        FNDIPL     0    BFSRCH    0
  RDREQ       0        BFSTLW     0    PURGWR    0
  WBSYID      0        WBSYWR     0    WBSYRD    0
  WRLSEO      0        WNOBFR     0    ERRORS 00000/00000
OSAM DB BUFFER POOL:  BSIZE ALL NBUF    12 OSM=   64K
  LCTREQ      0        NEWBLK     0    ALTREQ    0
  PURGRQ      0        FNDIPL     0    BFSRCH    0
  RDREQ       0        BFSTLW     0    PURGWR    0
  WBSYID      0        WBSYWR     0    WBSYRD    0
  WRLSEO      0        WNOBFR     0    ERRORS 00000/00000

VSAM DB BUFFER POOL:  ID XXXX BSIZE   1K TYPE D FX=N/N/N
  RRBA        0        RKEY       0    BFALT     0
  NREC        0        SYNC PT    0    NBUFS     8
  VRDS        0        FOUND      0    VWTS      0
  HSR-S       0        HSW-S      0    HS NBUFS  0
    HS R/W-FAIL 000000/000000    ERRORS 000000/000000
VSAM DB BUFFER POOL:  ID XXXX BSIZE   2K TYPE D FX=N/N/N
  RRBA        0        RKEY       0    BFALT     0
  NREC        0        SYNC PT    0    NBUFS     4
  VRDS        0        FOUND      0    VWTS      0
  HSR-S       0        HSW-S      0    HS NBUFS  0
    HS R/W-FAIL 000000/000000    ERRORS 000000/000000
VSAM DB BUFFER POOL:  ID XXXX BSIZE   8K TYPE D FX=N/N/N
  RRBA        0        RKEY       0    BFALT     0
  NREC        0        SYNC PT    0    NBUFS     4
```

```
   VRDS        0        FOUND      0    VWTS      0
   HSR-S       0        HSW-S      0    HS NBUFS  0
     HS R/W-FAIL 000000/000000    ERRORS 000000/000000
 VSAM DB BUFFER POOL:  BSIZE ALL VS=  48K    HS=  0K
   RRBA        0        RKEY       0    BFALT     0
   NREC        0        SYNC PT    0    NBUFS    16
   VRDS        0        FOUND      0    VWTS      0
   HSR-S       0        HSW-S      0    HS NBUFS  0
     HS R/W-FAIL 000000/000000    ERRORS 000000/000000
*92120/134346*
```

Example 6 for /DISPLAY POOL Command

Entry ET:

```
/DISPLAY POOL DBB
```

Response ET:

```
NAME SP#     CURR      MAX     GETS    FREES
PDIR 231      36K      36K        1        0
DDIR 231      28K      28K        1        0
SMB  231      61K      61K        1        0
BQEL 231       0K       0K        0        0
IEQE   0       0K       0K        0        0
EQEL 241       0K       0K        0        0
DDRE   0       0K       0K        0        0
CLASS = DBB      GLBL     125K  LCL       0K
CBT POOLS        GLBL    1187K  LCL     662K
*91051/161027*
```

Example 7 for /DISPLAY POOL Command

Entry ET:

```
/DISPLAY POOL DCC
```

Response ET:

```
NAME SP#     CURR      MAX     GETS    FREES
PCIB   0       0K       0K        0        0
VTCB 251     305K     305K        0        0
RECA   0      19K      19K        1        0
LGND 251      16K      16K        4        0
AHDR 251       0K       0K        0        0
IAFP 231       0K       0K        0        0
RACW 231       0K       0K        0        0
LUB  251       0K       0K        0        0
TIB  251      28K      28K        1        0
DESC 251       0K       0K        0        0
PF62 251       0K       0K        0        0
QAB  251       0K       0K        0        0
CBLK 251       0K       0K        0        0
CLASS = DCC      GLBL       0K  LCL     368K
CBT POOLS        GLBL    1187K  LCL     662K
*91051/161038*
```

Example 8 for /DISPLAY POOL Command

Entry ET:

```
/DISPLAY POOL FPDB
```

Response ET:

```
POOLNAME         CISIZE PBUF  SBUF   MAX CURRENT LK HITS VALID
V$$$CFSTRUCTURE1 04096  00100 00010 00500 00100  N  000%  NA
POOL4            04096  00010 00010 00500 00010  Y  NA    NA
```

Example 9 for /DISPLAY POOL Command

Entry ET:

```
/DISPLAY POOL MFP
```

Response ET:

```
MESSAGE FORMAT POOL:  SIZE     49152  SPACE     48576  DIRS     9011
 REQ1          2  I/O           2  DIR          4
 WAIT          2  FREE      48432  ERR          0
*91051/172502*
```

Example 10 for /DISPLAY POOL Command

Entry ET:

```
/DISPLAY POOL MFP CIOP MAIN
```

Response ET:

```
MESSAGE FORMAT POOL:  SIZE     49152  SPACE     48576  DIRS     9011
 REQ1          2  I/O           2  DIR          4
 WAIT          2  FREE      48432  ERR          0
CIOP BUFFER POOL:
 SIZE      96K  HIGH      96K  LIMIT     999K  OVERFLOW       0K
MAIN BUFFER POOL:
 SIZE      40K  FREE      40K  HIGH       5K
*91051/160913*
```

Example 11 for /DISPLAY POOL Command

Entry ET:

```
/DISPLAY POOL MFP STAT
```

Response ET:

```
MFBP BUFFER POOL:
 0170  00000000 00000002 00000002 00000000  00000004 00000000 00000000 00000002
 0190  00000000 00000000 00000000 00000002  00000000 00000000 00000000 00000000
 01B0  00000000 00000001
*91051/172508*
```

Example 12 for /DISPLAY POOL Command

Entry ET:

```
/DISPLAY POOL PSBP
```

Response ET:

```
PSBP BUFFER POOL:
 SIZE      80K  FREE      80K  HIGH       0K
DPSB BUFFER POOL:
 SIZE     160K  FREE     160K  HIGH       0K
*91051/161048*
```

Example 13 for /DISPLAY POOL Command

Entry ET:

```
/DISPLAY POOL QBUF
```

Response ET:

```
MESSAGE QUEUE POOL:   BFRS/SIZE 20/2112            SYS3
 ENQ     18  DEQ     18  CAN     52  WAIT      0  I/O     15  ERR   0
 QBLKS:  MAX # RECORDS AVAIL        5  CUR IN USE        3 =  60 %
 SMSGQ:  MAX # RECORDS AVAIL   262167  CUR IN USE        4 =   0 %
 LMSGQ:  MAX # RECORDS AVAIL   524293  CUR IN USE        2 =   0 %
*91051/172432*
```

Example 14 for /DISPLAY POOL Command

Entry ET:

```
/DISPLAY POOL QBUF STAT
```

Response ET:

```
QBUF BUFFER POOL:
 0054  00000000 0000005E 00000126 00000001  00000000 00000000 00000007 00000008
 0074  00000008 00000000 00000000 00000000  00000000 00000000 00000000 00000000
 0094  00000000 00000000 00000000 00000019  00000323 00000000 00000013 00000013
 00B4  00000035 00000003 00000004 00000002
*91051/172440*   SYS3
```

Example 15 for /DISPLAY POOL Command

Entry ET:

```
/DISPLAY POOL SUM
```

Response ET:

```
CBT POOLS        GLBL    1187K  LCL     662K
CLASS = OSAM     GLBL      28K  LCL       0K
CLASS = GEN      GLBL     724K  LCL      54K
CLASS = DEP      GLBL     216K  LCL       0K
CLASS = DISP     GLBL      88K  LCL     128K
CLASS = DBB      GLBL     125K  LCL       0K
CLASS = DCC      GLBL       0K  LCL     368K
CLASS = FP       GLBL       7K  LCL       0K
*91051/161003*
```

Example for /DISPLAY PROGRAM Command

Entry ET:

```
/DISPLAY PROGRAM APOL1 DBFSAMP3 BMP255
```

Response ET:

```
PROGRAM     TRAN      TYPE
APOL1        APOL11    TP
             APOL12
             APOL13
             APOL14
             APOL15
             APOL16
             APOL17
             APOL18
DBFSAMP3     FPSAMP1   FPM
BMP255       TRAN255   BMP
             TXCD255
             TXCD255W
*90288/140450*   SYS3
```

Entry ET:

```
/DISPLAY PROGRAM ALL
```

Response ET:

```
PROGRAM   TRAN       TYPE
AD2TP      TSTAD2A    TP   NOTINIT
           TSTAD2R1
APOL1      APOL11     TP   NOTINIT
           APOL12
A3270      A3270      TP   NOTINIT
           3270S
BMAAJK41   BHE4       TP   NOTINIT
           NQE4
```

```
           SHE4
BMP255     TRAN255    BMP  NOTINIT
BTAEJK01              BMP  NOTINIT
BTAPJK01              BMP  NOTINIT
BTAPJK11              BMP  NOTINIT
BTAPJK21              BMP  NOTINIT
BTGGJK01              BMP  NOTINIT
BTGGJK11              BMP  NOTINIT
BTGOJK01              BMP  NOTINIT
BTGRJK05              BMP  NOTINIT
CPGM1V0    CONV11V0   TP   NOTINIT
           CONV12V0
           CONV13V0
DBF#FPU0              FPU
DBFSAMP3   FPSAMP1    FPM
DBFSAMP4   FPSAMP2    TP   NOTINIT
DBFSAM22   TP1        FPM
DBFSAM99   TP2        TP   NOTINIT
DCLECHOS   DCL        TP   NOTINIT
DDLTBP04   TXCDBP04   FPM
DDLTBP05   TXCDBP05   FPM
DDLTBP06   TXCDBP06   FPM
DDLTBP07   TXCDBP07   FPM  NOTINIT
DDLTBP08   TXCDBP08   FPM  NOTINIT
DDLTBP09   TXCDBP09   FPM  NOTINIT
V2MPPA02   V2MRP01    TP   NOTINIT
           V2MRP02
V2MPPC01   V2SNS01    TP
           V2SRS02
V2MPPC02   V2MRS04    TP   NOTINIT
V2MPPP01   V2SNS02    TP   NOTINIT
           V2SRS03
           V2SRS04
V2MPPPP02  V2MRP03    TP   NOTINIT
           V2SRP01
WTCPLI     TESTPLI    TP   NOTINIT
*89184/142553*
```

Example for /DISPLAY PSB Command

Entry ET:

```
/DISPLAY PSB DDLTLM06 DDLTLM07
```

Response ET:

```
PSB-NAME    TRANCODE    RTCODE      DBD-NAME    ACCESS
DDLTLM06    TXCDLM06    TXCDLM06    MSDBLM01    R/W
                                    MSDBLM02    R/W
                                    MSDBLM03    R/W
                                    MSDBLM01    R/W
                                    MSDBLM04    R/W
                                    MSDBLM04    R/W
                                    MSDBLM05    R/W
                                    MSDBLM05    R/W
                                    MSDBLM06    R/W
                                    MSDBLM06    R/W
                                    MSDBLM06    R/W
                                    MSDBLM06    R/O
DDLTLM07    TXCDLM07    TXCDLM07    MSDBLM01    R/W
                                    MSDBLM02    R/W
                                    MSDBLM03    R/W
                                    MSDBLM01    R/O
                                    MSDBLM04    R/W
                                    MSDBLM04    R/W
                                    MSDBLM05    R/W
                                    MSDBLM05    R/W
```

```
                                        MSDBLM06    R/W
                                        MSDBLM06    R/W
                                        MSDBLM06    R/W
                                        MSDBLM06    R/O
                                        MSDBLM07    R/W
  *91068/111601*
```

Example for /DISPLAY PTERM Command

This example shows the display of all physical terminals that are being traced.

Entry ET:

```
/DISPLAY PTERM TRA
```

Response ET:

```
LIN/PTE  TYPE     ADDR  RECD ENQCT DEQCT   QCT   SENT
  2-   1 2740-II  ***      0     0     0     0      0STOPPED INOP NOTOPEN TRA
  2-   2 2740-II  ***      0     0     0     0      0STOPPED INOP NOTOPEN TRA
 11-   1 2740-II  ***      0     0     0     0      0STOPPED INOP NOTOPEN TRA
 11-   2 2740-II  ***      0     0     0     0      0STOPPED INOP NOTOPEN TRA
*89184/142950*
```

Examples for /DISPLAY Q Command

The following are examples of the /DISPLAY Q command.

Example 1 for /DISPLAY Q Command

Entry ET:

```
/DISPLAY Q
```

Response ET:

```
CLS CT  PTY CT  MSG CT  TRAN CT
  3        5        28      8
*90253/103811*
```

Entry ET:

```
/DISPLAY Q BALGRP
```

Response ET:

```
 BALGRP   NO.RGNS   MSG CT  ENQ COUNT  DEQ COUNT
DDLTM06         1        1          2          1
DDLTM07         1        0          0          0
*91068/112000*
```

Entry ET:

```
/DISPLAY Q BALGRP QCNT
```

Response ET:

```
BALGRP             GBLQCT
SMQFP1                  2
SMQFP2                  3
*96197/174618*
```

Entry ET:

```
/DISPLAY Q CLASS ALL
```

Response ET:

```
CLS     PTY CT   MSG CT  TRAN CT
  1        1        2        1
  2        1        2        1
  3        1        4        1
*90253/103855*
```

Entry ET:

```
/DISPLAY Q CLASS 1 2
```

Response ET:

```
CLS     PTY CT  MSG CT  TRAN CT
  1        1        2       1
  2        1        2       1
*90253/103913*
```

Entry ET:

```
/DISPLAY Q PRIORITY ALL
```

Response ET:

```
CLS     PTY     MSG CT  TRAN CT
  1        9        2       1
  2        1        2       1
  3        5        4       1
*90253/103945*
```

Entry ET:

```
/DISPLAY Q PRIORITY 10
```

Response ET:

```
CLS     PTY     MSG CT  TRAN CT
  4       10       18       3
*90253/1104016
```

Entry ET:

```
/DISPLAY Q CLASS 4 PRIORITY 10
```

Response ET:

```
CLS     PTY     MSG CT  TRAN CT
  4       10       18       3
*90309/122418*
```

Entry ET:

```
/DISPLAY Q TRANSACTION
```

Response ET:

```
CLS     PTY     MSG CT TRAN     PSBNAME
  3       10        3  CLOSE    DFSSAM05
  4       10        2  ADDPART  DFSSAM04
  4       10        4  PART     DFSSAM02
  5        7        1  ADDINV   DFSSAM04
  5       10        5  DLETINV  DFSSAM04
*90243/110324*
```

Entry ET:

```
/DISPLAY Q CLASS 3 5 TRANSACTION
```

Response ET:

```
CLS     PTY     MSG CT TRAN     PSBNAME
  3       10        3  CLOSE    DFSSAM05
  5        7        1  ADDINV   DFSSAM04
  5       10        5  DLETINV  DFSSAM04
*90243/110332*
```

Entry ET:

```
/DISPLAY Q CLASS 4 PRIORITY 10 11 TRANSACTION
```

Response ET:

```
CLS     PTY     MSG CT TRAN     PSBNAME
  4       10        2  ADDPART  DFSSAM04
  4       10        4  PART     DFSSAM02
*90243/110343*
```

Entry ET:

```
/DISPLAY Q CLASS 1 2 3 PRIORITY 1
```

Response ET:

```
CLS     PTY     MSG CT  TRAN CT
  1        1        2       1
  2        1        2       1
  3        1        4       1
*90253/104321*
```

Example 2 for /DISPLAY Q Command

Entry ET:

```
/DISPLAY QUEUE TRANSACTION
```

Response ET:

```
CLS    PTY    MSG  CT    TRAN     PSBNAME
 20      5         2     TPN1     DFSCPIC
 20      2         5     TPN2     SAAPSB1
 22      3         7     TPN3     SAAPSB2
*90332/114253*
```

Example for /DISPLAY QCNT Command

Entry ET:

```
/DISPLAY QCNT LTERM MSGAGE 5
```

Response ET:

```
QUEUENAME          QCNT-TOTAL      QCNT-AGED    TSTMP-OLD    TSTMP-NEW
TESTLTEA                    1              1    95280/132006 95280/132006
LTERMABC                    4              4    95274/083000 95275/091836
LTERMA                    100            100    95275/080000 95281/212224
*95290/132006*
```

Entry ET:

```
/DIS QCNT APPC MSGAGE 0
```

Response ET:

```
QUEUENAME          QCNT-TOTAL     QCNT-AGED    TSTMP-OLD    TSTMP-NEW
TMSTMP- SYS3               16            16    02192/080833 02192/080928
 LUNAME: L62MVS1
 TPNAME: TPNAME123
```

```
 TMSTMP- IMS2              12            12    02192/080620 02192/080658
  LUNAME: NETWRKID.L62IMS1
  TPNAME: ABCDEFGHIJKLMNOPQRSTUVWXYZ
*02192/081543*
```

Entry ET:

```
  /DISPLAY QCNT TRANSACTION MSGAGE 5
```

Response ET:

```
  QUEUENAME          QCNT-TOTAL      QCNT-AGED    TSTMP-OLD    TSTMP-NEW
  TRANA                      20             20    95200/132006 95280/132006
  TRANBBBB                    4              4    95274/083000 95275/091836
  TRANSACA                  220            220    95275/080000 95281/212224
  TRANSDDD                   13              0    95290/101455 95290/101456
  TRANSEEE                   55             13    95277/152118 95290/114317
  *95290/132006*
```

Entry ET:

```
  /DIS QCNT OTMA MSGAGE 0
```

Response ET:

```
 QUEUENAME          QCNT-TOTAL     QCNT-AGED    TSTMP-OLD    TSTMP-NEW
 TMSTMP- SYS3               24            24    02192/083745 02192/083745
  TMEMBER: VC5
  TPIPE: CSQ0000D
*02192/083757*
```

Examples for /DISPLAY RECOVERY Command

The following are examples of the /DISPLAY RECOVERY command.

Example 1 for /DISPLAY RECOVERY Command

In this example a /DISPLAY RECOVERY command is issued before a /RECOVER START for RCVTOKEN RECOV1 is issued.

Entry ET:

```
  /DISPLAY RECOVERY RCVTOKEN RECOV1
```

Response ET:

```
**** RECOVERY LIST INFORMATION *********************
TOKEN    STATUS   ERROR- REC TYPE  PROC  IC#  SOURCE
RECOV1   FORMING  N/A    N/A       N/A        N/A
**** RECOVERY LIST ENTRY INFORMATION ***************
DATABASE DATA SET  START OPTION  STATUS   AUTH SSID
DB23AR1            STAGLOBAL     NORMAL   NONE
DB23AR2            OFFLINE       NORMAL   NONE
DB23AR3            STAGLOBAL     NORMAL   NONE
DB23AR4            STALOCAL      NORMAL   NONE
DB23AR5            STALOCAL      NORMAL   NONE
DHVNTZ02 HIDAM     STAGLOBAL     NORMAL   NONE
DIVNTZ02 DBHVSAM1  OFFLINE       NORMAL   NONE
DXVNTZ02 XDLBT04I  STALOCAL      NORMAL   NONE
*03127/153515*
```

Example 2 for /DISPLAY RECOVERY Command

In the example below, a /DISPLAY RECOVERY command is issued before a /RECOVER START for RCVTOKEN RECOV1 is issued. Some of the database data sets in the recovery list are still authorized to two IMSs.

Entry ET:

```
/DISPLAY RECOVERY RCVTOKEN RECOV1
```

Response ET:

```
**** RECOVERY LIST INFORMATION *********************
TOKEN    STATUS   ERROR- REC TYPE  PROC  IC#  SOURCE
RECOV1   FORMING  N/A    N/A       N/A       N/A
**** RECOVERY LIST ENTRY INFORMATION ***************
DATABASE DATA SET  START OPTION  STATUS   AUTH SSID
DB23AR1            STAGLOBAL     NORMAL   NONE
DB23AR2            OFFLINE       NORMAL   NONE
DB23AR3            STAGLOBAL     NORMAL   NONE
DB23AR4            STALOCAL      NORMAL   NONE
DB23AR5            STALOCAL      NORMAL   NONE
DHVNTZ02 HIDAM     STAGLOBAL     NORMAL   IMS1
                                           IMS2
DIVNTZ02 DBHVSAM1  OFFLINE       NORMAL   IMS1
                                           IMS2
DXVNTZ02 XDLBT04I  STALOCAL      NORMAL   IMS1
                                           IMS2    *03127/153515*
```

Example 3 for /DISPLAY RECOVERY Command

In this example, a /DISPLAY RECOVERY command is issued after a /RECOVER START RCVTOKEN RECOV1 ERRORCONT command was issued when the IMS Database Recovery Facility is the recovery product being used. In this case, all logs have been processed and 3 of the 8 data sets have been restored.

Entry ET:

```
/DISPLAY RECOVERY RCVTOKEN RECOV1
```

Response ET:

```
**** RECOVERY LIST INFORMATION *********************
TOKEN    STATUS   ERROR- REC TYPE  PROC  IC#  SOURCE
RECOV1   STARTED  CONT   FULL      RTDB   0   PRI
**** RECOVERY PROGRESS INFORMATION *****************
TOKEN    PROGRESS INFORMATION     RCVTIME
RECOV1   0003 OF 0008 RESTORED    N/A
**** RECOVERY LIST ENTRY INFORMATION ***************
DATABASE DATA SET  START OPTION  STATUS   AUTH SSID
DB23AR1            STAGLOBAL     NORMAL   N/A
DB23AR2            OFFLINE       NORMAL   N/A
DB23AR3            STAGLOBAL     NORMAL   N/A
DB23AR4            STALOCAL      NORMAL   N/A
DB23AR5            STALOCAL      NORMAL   N/A
DHVNTZ02 HIDAM     STAGLOBAL     NORMAL   N/A
DIVNTZ02 DBHVSAM1  OFFLINE       NORMAL   N/A
DXVNTZ02 XDLBT04I  STALOCAL      NORMAL   N/A  *03127/153515*
```

Example 4 for /DISPLAY RECOVERY Command

In the following example, a /DISPLAY RECOVERY command is issued after a /RECOVER START ERRORCONT command was issued. Also a /RECOVER STOP command was issued for DHVNTZ02, and DB23AR2 failed during a recovery.

Entry ET:

```
/DISPLAY RECOVERY RCVTOKEN RECOV1
```

Response ET:

```
**** RECOVERY LIST INFORMATION *********************
TOKEN    STATUS   ERROR- REC TYPE  PROC  IC#  SOURCE
RECOV1   STARTED  CONT   FULL      RTDB   0   PRI
**** RECOVERY PROGRESS INFORMATION *****************
TOKEN    PROGRESS INFORMATION     RCVTIME
RECOV1   2000.251 07:57:00.3      N/A
```

```
**** RECOVERY LIST ENTRY INFORMATION ***************
DATABASE DATA SET  START OPTION  STATUS   AUTH SSID
DB23AR1            STAGLOBAL     NORMAL   N/A
DB23AR2            OFFLINE       FAILED   N/A
DB23AR3            STAGLOBAL     NORMAL   N/A
DB23AR4            STALOCAL      NORMAL   N/A
DB23AR5            STALOCAL      NORMAL   N/A
DHVNTZ02 HIDAM     STAGLOBAL     STOPPED  N/A
DIVNTZ02 DBHVSAM1  OFFLINE       NORMAL   N/A
DXVNTZ02 XDLBT04I  STALOCAL      NORMAL   N/A  *03127/153515*
```

Example 5 for /DISPLAY RECOVERY Command

In the example below, a /DISPLAY RECOVERY ALL command is issued after three lists are created using the IMS Database Recovery Facility recovery product. In this case the IMS Database Recovery Facility has processed 76 of the 126 logs that will need to be read. One of the lists (RECOV2) is undergoing TSR.

Entry ET:

```
/DISPLAY RECOVERY ALL
```

Response ET:

```
**** RECOVERY LIST INFORMATION *********************
TOKEN    STATUS   ERROR- REC TYPE  PROC  IC#  SOURCE
RECOV1   FORMING  N/A    N/A       N/A        N/A
RECOV2   STARTED  CONT   TSR       RTDB   0   PRI
RECOV3   FORMING  N/A    N/A       N/A        N/A
**** RECOVERY PROGRESS INFORMATION *****************
TOKEN    PROGRESS INFORMATION     RCVTIME
RECOV2   0076 OF 0125 LOGS READ   2003.127 08:30:00.0
**** RECOVERY LIST ENTRY INFORMATION ***************
DATABASE DATA SET  START OPTION  STATUS   AUTH SSID
DB23AR1            STAGLOBAL     NORMAL   N/A
DB23AR2            OFFLINE       NORMAL   N/A
DB23AR3            STAGLOBAL     NORMAL   N/A
DB23AR4            STALOCAL      NORMAL   N/A
DB23AR5            STALOCAL      NORMAL   N/A
DHVNTZ02 HIDAM     STAGLOBAL     NORMAL   N/A
DIVNTZ02 DBHVSAM1  OFFLINE       NORMAL   N/A
DXVNTZ02 XDLBT04I  STALOCAL      NORMAL   N/A  *03127/153515*
```

Example 6 for /DISPLAY RECOVERY Command

In the example below, a /DISPLAY RECOVERY command is issued after three lists are created. No other parameters are specified.

Entry ET:

```
/DISPLAY RECOVERY ALL
```

Response ET:

```
**** RECOVERY LIST INFORMATION *********************
TOKEN    STATUS   ERROR- REC TYPE  PROC  IC#  SOURCE
RECOV1   FORMING  N/A    N/A       N/A        N/A
RECOV2   FORMING  N/A    N/A       N/A        N/A
RECOV3   FORMING  N/A    N/A       N/A        N/A
*03127/153515*
```

Example 7 for /DISPLAY RECOVERY Command

In the example below, a /DISPLAY RECOVERY ALL command is issued when no recovery lists exist.

Entry ET:

```
/DISPLAY RECOVERY ALL
```

Response ET:

```
**** RECOVERY LIST INFORMATION *********************
TOKEN    STATUS   ERROR- REC TYPE  PROC  IC#  SOURCE
**** RECOVERY LIST INFORMATION *********************
TOKEN    STATUS   ERROR- REC TYPE  PROC  IC#  SOURCE
NO LISTS
*03127/153515*
```

Example 8 for /DISPLAY RECOVERY Command

In the example below, a /DISPLAY RECOVERY RCVTOKEN RECOV3 is issued when the RCVTOKEN does not exist.

Entry ET:

```
/DISPLAY RECOVERY RECVTOKEN RECOV3
```

Response ET:

```
**** RECOVERY LIST INFORMATION *********************
TOKEN    STATUS   ERROR- REC TYPE  PROC  IC#  SOURCE
**** RECOVERY LIST INFORMATION *********************
TOKEN    STATUS   ERROR- REC TYPE  PROC  IC#  SOURCE
RECOV3   UNKNOWN  N/A    N/A   *03127/153515*
```

Example for /DISPLAY RTCODE Command

Entry ET:

```
/DISPLAY RTCODE ALL
```

Response ET:

```
RTCODE      PROGRAM     STATUS
DCL         EMHPSB      NOTSCHED,STOPPED
EMHCOBOL    EMHCOBOL    NOTSCHED,STOPPED
EMHPLI      EMHPLI      NOTSCHED,STOPPED
EMHTX       EMHPSB      NOTSCHED,STOPPED
TXCDLM06    DDLTLM06    STOPPED
TXCDLM07    DDLTLM07    STOPPED
*91068/114840*
```

Examples of /DISPLAY SHUTDOWN STATUS Command

The following are examples of the /DISPLAY SHUTDOWN STATUS command.

Example 1 for /DISPLAY SHUTDOWN STATUS Command

Entry ET:

```
/DISPLAY SHUTDOWN STATUS
```

Response ET:

```
DFS134 SHUTDOWN CHECKPOINT NOT IN PROGRESS,
CANNOT PROCESS COMMAND
```

Entry ET:

```
/DISPLAY SHUTDOWN STATUS
```

Response ET:

```
SYSTEM PURGING

TERMINAL  USER  STATUS
   5-   1          INPUT  IN PROCESS
   5-   2          INPUT  IN PROCESS
  11-   1          OUTPUT IN PROCESS
TERMINAL  USER  STATUS
```

```
LINK   10          OUTPUT IN PROCESS
    MSG-IN    2   MSG-OUT   2
  MASTER ACTIVE
*91111/222226*
```

Entry ET:

```
/DISPLAY SHUTDOWN STATUS
```

Response ET:

```
TERMINAL  USER  STATUS
NSLUTP6   SPOOLA   AWAITING RESPONSE
NSLUTP6   SPOOLB     5 MSGS IN QUEUE
NSLUTP6   SPOOLC   INPUT  IN PROCESS
NSLUTP6   SPOOLD   OUTPUT IN PROCESS
*91111/222226*
```

Example 2 for /DISPLAY SHUTDOWN STATUS Command

Entry ET:

```
/DISPLAY SHUTDOWN STATUS
```

Response ET:

```
TERMINAL  USER STATUS
   5-   1         INPUT IN PROCESS
   5-   2         INPUT IN PROCESS
  11-   1         OUTPUT IN PROCESS
TERMINAL  USER STATUS
LINK   10         OUTPUT IN PROCESS
    MSG-IN   2  MSG-OUT   2
  MASTER ACTIVE
CPI TRAN TRNCODE1 ACTIVE IN REGID 1
CPI TRAN TRNCODE2 ACTIVE IN REGID 4
IMSLU=L62IMS    #APPC-CONV=          8 ENABLED
LUNAME            STATUS
LUNAME1               3 CONVERSATIONS(S) IN PROCESS
LUNAME1               1 OUTPUT IN PROCESS
LUNAME2               4 CONVERSATIONS(S) IN PROCESS
*90332/114253*
```

Entry ET:

```
/DISPLAY SHUTDOWN STATUS
```

Response ET:

```
TERMINAL   SUBPOOL STATUS
   5-  1           INPUT IN PROCESS
   5-  2           INPUT IN PROCESS
  11-  1           OUTPUT IN PROCESS
TERMINAL   SUBPOOL STATUS
LINK  10           OUTPUT IN PROCESS
   MSG-IN    2   MSG-OUT   2
 MASTER ACTIVE
CPI TRAN TRNCODE1 ACTIVE IN REGID1
CPI TRAN TRNCODE2 ACTIVE IN REGID4
IMSLU=SYS1.IMSLUNME      #APPC-CONV=       8  ENABLED
LUNAME           STATUS
NETWORK1.LUNAME1      3 CONVERSATION(S) IN PROCESS
NETWORK1.LUNAME1      1 OUTPUT IN PROCESS
NID2.LUNAME2          4 CONVERSATION(S) IN PROCESS
*92232/114253*
```

Example 3 for /DISPLAY SHUTDOWN STATUS Command

Entry ET:

```
/DISPLAY SHUTDOWN STATUS
```

Response ET:

```
TERMINAL  USER     STATUS
   1-   1          INPUT  IN PROCESS
TERMINAL  USER     STATUS
  NO OUTPUTTING LINES
    MSG-IN    1     MSG-OUT   0
  MASTER ACTIVE
OTMA PHASE=2
COMMIT 0  TMEMBER=CLIENT1          TPIPE=TPIPE1
*94298/174604*
```

Examples for /DISPLAY STATUS Command

The following are examples of the /DISPLAY STATUS command.

Example 1 for /DISPLAY STATUS Command

Entry ET:

```
/DISPLAY STATUS
```

Response ET:

```
**TRAN****PSBNAME
STATUS UNRESTRICTED
**DATABASE**
BMACC1    NOTOPEN,NOTINIT,STOPPED
DEDB01
-AREA01     RECOVERY-NEEDED
DEDEB03
-AREA02
  ADS01     PRE-OPEN FAILED
DEDB04    NOTOPEN,STOPPED
-AREA01     RECOVERY-NEEDED
  ADS01     UNAVAILABLE
**PROGRAM***
HIMASN01  STOPPED
SWITCH    STOPPED
**NODE****
CT3275    DISCONNECTED
CT3277A   DISCONNECTED
CT3277B   SHUT
W612      DISCONNECTED
**LINE******
4         STOPPED,IDLE
5         STOPPED,IDLE,NOTOPEN
8         STOPPED,IDLE
10        STOPPED,IDLE,NOTOPEN
**PTERM*****
3         1 INOP, STOPPED
3         2 INOP, STOPPED
4         2 INOP, PSTOPPED
5         1 INOP, STOPPED
5         2 INOP, STOPPED
6         1 INOP, STOPPED
**LTERM*****
LU6A      STOPPED
VA01      QLOCK
**CLASS*****
2         STOPPED
**RTCODE****
STATUS UNRESTRICTED
**LINK******
1         PSTOPPED,IDLE,NOTOPEN
3         PSTOPPED,IDLE,COLD
4         PSTOPPED,IDLE,N/A
5         PSTOPPED,IDLE,NOTOPEN
**MSNAME****
```

```
STATUS UNRESTRICTED
**USER******
STATUS UNRESTRICTED
**LUNAME**TPNAME****
STATUS UNRESTRICTED
*92198/132223*
```

Example 2 for /DISPLAY STATUS Command

Entry ET:

```
/DISPLAY STATUS DATABASE
```

Response ET:

```
**DATABASE**
DEDB01
-AREA01     STOPPED
DEDB02
-AREA02     NOTOPEN
-AREA03
  ADS03     PRE-OPEN FAILED
DEDB03
-AREA05     STOPPED,NOTOPEN
  ADS03     PRE-OPEN FAILED
  ADS08     UNAVAILABLE
DEDB04    STOPPED
DEDB05    NOTOPEN,NOTINIT
-AREA01     STOPPED
-AREA08     STOPPED,NOTOPEN
DEDB06    STOPPED
-AREA03
  ADS03     UNAVAILABLE
DEDB0777  NOTOPEN,LOCK
-AREA0555   STOPPED,NOTOPEN
  ADS03333  SEVERE ERROR
  ADS08888  UNAVAILABLE
*90263/092128*
```

Example 3 for /DISPLAY STATUS Command

Entry ET:

```
/DISPLAY STATUS LUNAME
```

Response ET:

```
**LUNAME/TPNAME**
LU2
-TPN2               STO
LU5
-TPNAME1234567890+  STO
 1234567890
LU3                 STO-INP
LU4                 STO-OUTP
LU1                 STO-INP, STO-OUTP
*95229/155100*
```

Entry ET:

```
/DISPLAY STATUS LUNAME
```

Response ET:

```
**LUNAME/TPNAME**
LU2                 STO-INP, STO-OUTP
-TPN2               STO
LU5                 STO-INP, STO-OUTP
-TPN5               STO
-TPNAME1234567890+  STO
 1234567890
```

```
  LU3                 STO-INP, STO-OUTP
  LU4                 STO-INP, STO-OUTP
  LU1                 STO-INP, STO-OUTP
  FUTURE INPUT:  STO
  FUTURE OUTPUT: STO
  *95229/155146*
```

Example 4 for /DISPLAY STATUS Command

Entry ET:

```
  /DISPLAY STATUS NODE
```

Response ET:

```
  **NODE******
  NLUTP6S   DISCONNECTED
  NLU1      STOPPED, DISCONNECTED
  NLUTP6P
  -A        HELD
  -N/A      STOPPED, DISCONNECTED
  -E        FORCE
  -C
  *90231/040059*
```

Entry ET:

```
  /DISPLAY STATUS RTCODE
```

Response ET:

```
  **RTCODE****
  DCL       STOPPED
  EMHCOBOL  STOPPED
  EMHPLI    STOPPED
  EMHTX     STOPPED
  TXCDBP04  STOPPED
  TXCDBP05  STOPPED
  TXCDBP06  STOPPED
  *91068/114900*
```

Example 5 for /DISPLAY STATUS Command

Entry ET:

```
  /DISPLAY STATUS TMEMBER
```

Response ET:

```
 **TMEMBER/TPIPE**
CLIENT1
-TPIPE1             STO
*94168/095431*
```

Example 6 for /DISPLAY STATUS Command

Entry ET:

```
  /DISPLAY STATUS TRANSACTION
```

Response ET:

```
  **TRAN****PSBNAME
  TPN4      DFSCPIC  PUR
  TPN5      DFSCPIC  STOPPED
  TPN6      DFSCPIC  I/O PREVEN
  TPN7      DFSCPIC  USTOPPED
  TRAN1     LU2PGM1  PSTOPPED
  TRAN2     LU2PGM2  PSTOPPED
  *90332/114253*
```

Example for /DISPLAY STRUCTURE Command

Entry ET:

```
/DISPLAY STRUCTURE ALL
```

Response ET:

```
STRUCTURE NAME    TYPE  STATUS
MSGQ1             MSGQ  CONNECTED, AVAILABLE, SHUTDOWN-STRCHKPT
EMHQ1             EMHQ  CONNECTED, AVAILABLE, IN-OVERFLOW
*95200/170817*
```

Entry ET:

```
/DISPLAY STRUCTURE MSGQ111 EMHQ1
```

Response ET:

```
STRUCTURE NAME   TYPE  STATUS
MSGQ111 IS INVALID
EMHQ1            EMHQ  DISCONNECTED
*95201/121800
```

Examples for /DISPLAY SUBSYS Command

The following are examples of the /DISPLAY SUBSYS command.

Example 1 for /DISPLAY SUBSYS Command

Entry ET:

```
/DISPLAY SUBSYS ALL
```

Response ET:

```
 SUBSYS   CRC REGID PROGRAM  LTERM   STATUS
 SSTR     ?                         CONN
                 1 DDLTLM17 PTERM01 CONN, ACTIVE
                 2 DDLTLM06 PTERM02 CONN
 *89202/065933*
```

Example 2 for /DISPLAY SUBSYS Command

Entry ET:

```
/DISPLAY SUBSYS XXX1 XXX3
```

Response ET:

```
SUBSYS   CRC REGID PROGRAM  LTERM   STATUS
XXX1     1                         CONN
XXX3     3                         CONN
*90083/154241*
```

Example for /DISPLAY SYSID Command

Entry ET:

```
/DISPLAY SYSID TRANSACTION SKS1 SKS2 SKS3 SKS4
```

Response ET:

```
TRAN      PSBNAME   RID    LID
SKS1      DFSDDLT1   21     11
SKS2      DFSDDLT2   NA     11
SKS3      DFSDDLT3   NA     11
SKS4      DFSDDLT4   32     12
*89184/142436*
```

Example for /DISPLAY TIMEOVER Command

Entry ET:

```
/DISPLAY TIMEOVER 10
```

Response ET:

```
NODE      USER   TYPE  LAST OUTPUT START TIME
SLU1A     N/A       SLU1  88118/095914
LU6NDPC   LU6SPC    LUT6  88118/095921
SLU1C     N/A       SLU1  88118/095929
SLU1B     N/A       SLU1  88118/095937
SLU1D     N/A       SLU1  88118/095943
LU6NDPA   LU6SPA    LUT6  88118/095949
*88118/101150*
```

Examples for /DISPLAY TMEMBER Command

The following are examples of the /DISPLAY TMEMBER command.

Example 1 for /DISPLAY TMEMBER Command

Entry ET:

```
/DISPLAY TMEMBER ALL
```

Response ET:

```
GROUP/MEMBER      XCF-STATUS   USER-STATUS    SECURITY
APPL8             ACTIVE       SERVER         FULL
CLIENT1           ACTIVE       ACCEPT TRAFFIC
*94165/170450*
```

Example 2 for /DISPLAY TMEMBER Command

Entry ET:

```
/DISPLAY TMEMBER CLIENT1 TPIPE ALL
```

Response ET:

```
MEMBER/TPIPE          ENQCT      DEQCT        QCT STATUS
CLIENT1                                          TRA
-TPIPE1                  0          0          0 TRA,STO
-TPIPE2                  2          2          0 TRA,STO
-TPIPE3                  1          0          1 TRA,STO
*94165/170756*
```

Examples for /DISPLAY TRACE Command

The following are examples of the /DISPLAY TRACE command.

Example 1 for /DISPLAY TRACE Command

Entry ET:

```
/DISPLAY TRACE ALL
```

Response ET:

```
IMS ACTIVE TRACES

LINE      TYPE
     1  CONSOLE
     4  3270 R

LINK    PARTNER
    2    AC

NODE        TYPE   USERS
 SLUP1      SLUP
```

```
 LU6NDPA    LUT6   N/A       N/A       N/A
 VAT11      3277

NO PSB TRACES FOUND

MONITOR IS INACTIVE

TRAN CODE     PSB-NAME
 ADDINV       DFSSAM04

PROGRAM NAME
 APOL1
 DEBS
 TACP1
  TABLE    TRACE   LOGGING  ENTRIES VOLUME
   RETR     ON       N/A   128/PST  N/A
   DL/I             OFF      2268
     DL/I   ON                      HIGH
     LOCK   ON                      MEDIUM
     PI     ON                      LOW
   LATC     ON       OFF      1512  MEDIUM
   DISP     ON       OFF      1260  LOW
   SCHD     OFF      OFF       630  OFF
   SUBS     OFF      OFF      1008  OFF
   DLOG     OFF      OFF       756  OFF
   FAST     OFF      OFF       252  OFF
   STRG     ON       OFF      1260  HIGH
   IDC0     ON       OFF      2268  HIGH
   LUMI     ON       OFF      1008  LOW

  PI TRACE IS ACTIVE     OPTIONS:  NOT LOGGED  NOTIME  VOLUME: MEDIUM

  NO TRAP TRACES FOUND

  IMS EXTERNAL TRACE IS  USABLE

 XTRC DDNAME    ALLOC STATUS             STATUS
  DFSTRA01      DYNA    UNALLOCATED      CLOSED

  DFSTRA02      DYNA    UNALLOCATED      CLOSED

  DFSTRA0T      DYNA    UNALLOCATED      CLOSED



VTAM I/O TIMEOUT FACILITY IS INACTIVE
TCO IS NOT ACTIVE

  LUNAME    TPNAME            TYPE
  NO LUNAME TRACES FOUND

  OSAMGTF TRACE IS INACTIVE
  *94060/100733*
```

Example 2 for /DISPLAY TRACE Command

Entry ET:

```
  /DISPLAY TRACE LINE LINK NODE
```

Response ET:

```
IMS ACTIVE TRACES

LINE      TYPE
     1  CONSOLE
    30  3270 R
    30  3270 R
```

```
LINK    PARTNER
    4    RF
NODE        TYPE    USERS
 L3270A     3277
 LU6NDPA    LUT6    N/A    N/A     N/A
LEVEL: 4    MODULE: ALL
*90290/143519*
```

Example 3 for /DISPLAY TRACE Command

Entry ET:

```
  /DISPLAY TRACE LUNAME
```

Response ET:

```
  IMS ACTIVE TRACES

  LUNAME    TPNAME
   LUNAME3                      INPUT
   LUNAME3  TPNAME1             OUTPUT
   LUNAME3  TPNAME7890123456+ OUTPUT
            789END
   LUNAME4                      INPUT
  *90332/114253*
```

Entry ET:

```
/DISPLAY TRACE LUNAME
```

Response ET:

```
IMS ACTIVE TRACES

LUNAME/TPNAME
NET1.LU1            INP
-TPN1
NET2.LU1            OUTP
-TPN2               OUTP
LU3                 INP, OUTP
-TPN3               OUTP
-TPN4               OUTP
NET4.LU3            OUTP
NET5.LU5            INP
*93069/114435*
```

Example 4 for /DISPLAY TRACE Command

Entry ET:

```
  /DISPLAY TRACE OSAMGTF
```

Response ET:

```
IMS ACTIVE TRACES
OSAMGTF TRACE IS INACTIVE
*92107/090819
```

Entry ET:

```
  /TRACE SET ON OSAMGTF
```

Response ET:

```
DFS058I 09:08:32 TRACE COMMAND COMPLETED
```

Entry ET:

```
  /DISPLAY TRACE OSAMGTF
```

Response ET:

```
IMS ACTIVE TRACES
OSAMGTF TRACE IS ACTIVE
*92107/090841*
```

Example 5 for /DISPLAY TRACE Command

Entry ET:

```
/DISPLAY TRACE MONITOR
```

Response ET:

```
IMS ACTIVE TRACES
MONITOR IS ACTIVE:   LA  SCHD  APMQ  APDB
   APDB dbname/partition-name/area-name ...
   REGION reg# ... region-name ...
   INTERVAL #seconds EXPIRING hh:mm:ss.tt
*98029/114114*
```

Example 6 for /DISPLAY TRACE Command

Entry ET:

```
/TRACE SET ON TABLE QMGR
/TRACE SET ON TABLE SQTT
/DISPLAY TRACE TABLE
```

Response ET:

```
IMS ACTIVE TRACES
TABLE    TRACE   LOGGING  ENTRIES VOLUME
 RETR     ON       N/A   128/PST  N/A
 DL/I             OFF      2268
   DL/I   ON                      MEDIUM
   LOCK   ON                      MEDIUM
   PI     OFF                     OFF
 LATC     ON       OFF      1512  MEDIUM
 DISP     ON       OFF      1260  MEDIUM
 SCHD     ON       OFF       630  MEDIUM
 SUBS     OFF      OFF      1008  OFF
 DLOG     ON       OFF       756  MEDIUM
 FAST     OFF      OFF       252  OFF
 STRG     ON       OFF      1260  MEDIUM
 IDC0     OFF      OFF      2268  OFF
 LUMI     OFF      OFF      1008  OFF
 OTMT     OFF      OFF      1008  OFF
 QMGR     ON       OFF      1008  MEDIUM
 SQTT     ON       OFF      1008  MEDIUM
 *96225/153244*
```

Example 7 for /DISPLAY TRACE Command

Entry ET:

```
/DISPLAY TRACE TCO
```

Response ET:

```
IMS ACTIVE TRACES

TCO IS NOT ACTIVE
*89100/170509*
```

Example 8 for /DISPLAY TRACE Command

Entry ET:

```
/DISPLAY TRACE TIMEOUT
```

Response ET:

```
 IMS ACTIVE TRACES

 VTAM I/O TIMEOUT FACILITY IS INACTIVE

 *89033/170200*
```

Example 9 for /DISPLAY TRACE Command

Entry ET:

```
  /DISPLAY TRACE TIMEOUT
```

Response ET:

```
 IMS ACTIVE TRACES

 VTAM I/O TIMEOUT FACILITY IS ACTIVE
        OPTION = NONE   TIMEOUT VALUE = 0

 *89110/131429*
```

Example 10 for /DISPLAY TRACE Command

Entry ET:

```
  /DISPLAY TRACE TIMEOUT
```

Response ET:

```
 IMS ACTIVE TRACES

 VTAM I/O TIMEOUT FACILITY IS ACTIVE
       OPTION = AUTO    TIMEOUT VALUE = 10

 *89033/170200*
```

Example 11 for /DISPLAY TRACE Command

Entry ET:

```
  /DISPLAY TRACE TIMEOUT
```

Response ET:

```
 IMS ACTIVE TRACES

 VTAM I/O TIMEOUT FACILITY IS ACTIVE
       OPTION = MSG     TIMEOUT VALUE = 60

 *89033/131630*
```

Example 12 for /DISPLAY TRACE Command

Entry ET:

```
  /DISPLAY TRACE TMEMBER
```

Response ET:

```
IMS ACTIVE TRACES

TMEMBER/TPIPE       TYPE
CLIENT1
-TPIPE1             TRA
*94168/095325*
```

Example 13 for /DISPLAY TRACE Command

Entry ET:

```
  /DISPLAY TRACE XTRC
```

Response ET:

```
IMS ACTIVE TRACES

IMS EXTERNAL TRACE IS  USABLE
     SYS3
XTRC DDNAME   ALLOC STATUS             STATUS
 DFSTRA01     DYNA    UNALLOCATED      CLOSED
 DFSTRA02     DYNA    UNALLOCATED      CLOSED
 DFSTRA0T     DYNA    UNALLOCATED      CLOSED

*91091/124215*
```

Example 14 for /DISPLAY TRACE Command

Entry ET:

```
/DIS TRACE EXIT
```

Response ET:

```
55/DIS TRACE EXIT

IEE600I REPLY TO 55 IS;/DIS TRACE EXIT
DFS000I     IMS ACTIVE TRACES   IMS3
DFS000I          IMS3
DFS000I        EXIT       FUNC   STATUS    IMS3
DFS000I      DFSMSCE0     TRBT     ON          IMS3
DFS000I      DFSMSCE0     TRVT     ON          IMS3
DFS000I      DFSMSCE0     TR62     OFF         IMS3
DFS000I      DFSMSCE0     TROT     OFF         IMS3
DFS000I      DFSMSCE0     LRTR     ON          IMS3
DFS000I      DFSMSCE0     LRLT     ON          IMS3
DFS000I      DFSMSCE0     LRDI     ON          IMS3
DFS000I      DFSMSCE0     LRIN     ON          IMS3
DFS000I      DFSMSCE0     PRCH     ON          IMS3
DFS000I      DFSMSCE0     PRIS     N/A         IMS3
DFS000I     *99096/103002*   IMS3
56 DFS996I *IMS READY*  IMS3
```

Response ET:

```
39/DIS TRACE EXIT
IEE600I REPLY TO 39 IS;/DIS TRACE EXIT
DFS000I     IMS ACTIVE TRACES   IMS3
DFS000I          IMS3
DFS000I        EXIT       FUNC   STATUS    IMS3
DFS000I      DFSMSCE0     TRBT     N/A         IMS3
DFS000I      DFSMSCE0     TRVT     N/A         IMS3
DFS000I      DFSMSCE0     TR62     N/A         IMS3
DFS000I      DFSMSCE0     TROT     N/A         IMS3
DFS000I      DFSMSCE0     LRTR     N/A         IMS3
DFS000I      DFSMSCE0     LRLT     N/A         IMS3
DFS000I      DFSMSCE0     LRDI     N/A         IMS3
DFS000I      DFSMSCE0     LRIN     N/A         IMS3
DFS000I      DFSMSCE0     PRCH     N/A         IMS3
DFS000I      DFSMSCE0     PRIS     N/A         IMS3
DFS000I     *99105/222611*   IMS3
46 DFS996I *IMS READY*  IMS3
```

Examples for /DISPLAY TRACKING STATUS Command

The following are examples of the /DISPLAY TRACKING STATUS command.

Example 1 for /DISPLAY TRACKING STATUS Command

The following is an example of a /DISPLAY TRACKING STATUS entered from a tracking subsystem in Los Angeles. The two active systems in San Jose are named IMSA and IMSC. IMSA has an XRF alternate named IMSB. IMSC has an XRF alternate

named IMSD. The Coordinated Universal Time is 23:32:09. The local Los Angeles time is 15:32:09. OFR is also in progress and has started processing log data up through 14:11:34 local time.

Entry ET:

```
/DIS TRACKING STATUS
```

Response ET:

```
**** TRACKING SUBSYSTEM ****************************************
IMSID  READINESS-LEVEL  GSG-NAME  SG-NAME   SG-STATUS
IMST   RECOVERY         IMSGSG1   STLSITE2  IDENTIFIED
**** TRACKING STATUS:  TRACKING
NUMBER OF ACTIVE SUBSYSTEMS:                        4
**** TRACKING ACTIVITY *****************************************
NAME      TYPE    RECEIVED-LOG  ROUTED-LOG  NOT-ROUTED  STATUS
IMSD      ONLINE  N/A           N/A                N/A  CONV-ACT
IMSC      ACTIVE  15:31:00      15:29:54            2K  CONV-ACT
IMSB      ONLINE  N/A           N/A                N/A  CONV-ACT
IMSA      ACTIVE  15:31:10      15:29:47            3K  CONV-ACT
**** GAPS ******************************************************
NAME      LOG-SEQ-FIRST     IN-GAP  RECEIVED  NOT-ROUTED STATUS
IMSC      0000000000007D31      2K        0K         N/A WAITING
**** ISOLATED LOG SENDER STATUS ********************************
SYSTEM NAME OF TRANSPORT MANAGER SUBSYSTEM:  TMPAR
NUMBER OF CONVERSATIONS:                            1
**** ONLINE FORWARD RECOVERY STATUS ****************************
NUMBER OF STREAMS BEING PROCESSED:                  0
OFR ROUTING TIME:                    2002.090 14:11:34
**** MILESTONE INDEX STATUS ************************************
CURRENT MILESTONE INDEX:                           38
RESTART MILESTONE INDEX:                           37
**** DLI TRACKING STATUS ***************************************
TRACKING PSTS:                                      0
USAGE OF TRACKING PSTS:                            0%
BACKLOG OF REDO RECORDS IN DATASPACE:              0K
**** FP TRACKING STATUS ****************************************
TRACKING PSTS:                                      0
CURRENT LOG VOLUME IN DATASPACE:                   0K
MAX LOG VOLUME IN DATASPACE:                       0K
*02090/153209*
```

Example 2 for /DISPLAY TRACKING STATUS Command

The following is an example of a /DISPLAY TRACKING STATUS command entered on an active system IMSC, located in San Jose.

See “Example 1 for /DISPLAY ACTIVE Command” on page 282 for a description of the sample RSR environment.

Entry ET:

```
/DIS TRACKING STATUS
```

Response ET:

```
**** ACTIVE SUBSYSTEM ***************************************************
IMSID  GSG-NAME  SG-NAME   SG-STATUS
IMSC   ACCTGRP1  SITESJ    IDENTIFIED
**** ACTIVE'S CONVERSATIONS WITH LOG ROUTERS ****************************
IMSID  RECORDS-NOT-SENT
IMST                1K SENDING LOGS
*91317/150416*
```

Examples for /DISPLAY TRANSACTION Command

The following are examples of the /DISPLAY TRANSACTION command.

Example 1 for /DISPLAY TRANSACTION Command

Entry ET:

```
/DISPLAY TRANSACTION ALL
```

Response ET:

```
   TRAN   CLS ENQCT   QCT   LCT  PLCT CP NP LP SEGSZ SEGNO PARLM RC
TPN1       22    14     7     5     0  8  5  8     0     0     0  5
    PSBNAME: DFSCPIC
TPN2       23    16     0     0     0  0  0  0     0     0     0  0
    PSBNAME: DFSCPIC
    STATUS: PUR
TPN2       24     0     0     0     0  0  0  0     0     0     0  0
    PSBNAME: DFSCPIC
    STATUS: STOP
TRAN1       4     0     0    10     5  8  8 10     0     0     2  2
    PSBNAME: LU2PGM1
TRAN2       7     0     0 65535 65535  8  8  8     0     0     0  0
    PSBNAME: LU2PGM2
    STATUS: PSTOP
*90332/114253*
```

Example 2 for /DISPLAY TRANSACTION Command

Entry ET:

```
/DISPLAY TRANSACTION SKS1 SKS2 SKS3 SKS4 SKS5 SKS6 SKS7
```

Response ET:

```
   TRAN   CLS ENQCT   QCT   LCT  PLCT CP NP LP SEGSZ SEGNO PARLM RC
SKS1        1     0     0 65535 65535  8  8  8     0     0     0  0
    PSBNAME: DFSDDLT1
    STATUS:  TRA
SKS2        2     0     0 65535 65535  8  8  8     0     0     0  0
    PSBNAME: DFSDDLT2
SKS3        3     0     0 65535 65535  8  8  8     0     0     0  0
    PSBNAME: DFSDDLT3
    STATUS: STOP,TRA
SKS4        4     0     0 65535 65535  8  8  8     0     0     0  0
    PSBNAME: DFSDDLT4
    STATUS: STOP
SKS5        5     0     0 65535 65535  8  8  8     0     0     0  0
    PSBNAME: DFSDDLT5
SKS6        6     0     0 65535 65535  8  8  8     0     0     2  0
    PSBNAME: DFSDDLT6
    STATUS: BAL(  2)
SKS7        7     0     0 65535 65535  8  8  8     0     0     1  0
    PSBNAME: DFSDDLT7
    STATUS: BAL( 2)
*89184/142345*
```

Example 3 for /DISPLAY TRANSACTION Command

Entry ET:

```
/DISPLAY TRANSACTION TRANA TRANB TRANC QCNT
```

Response ET:

```
TRAN         GBLQCT
TRANA             0
TRANB          1000
TRANC            45
*95200/170817*
```

Example 4 for /DISPLAY TRANSACTION Command

Entry ET:

```
/DISPLAY TRAN ALL QCNT
```

Response ET:

```
TRAN         GBLQCT  AFFINITY
TRAN1234              1524  IMS1
TRAN1234               3  IMSA
TRAN1234               14
*00305/103034*
```

Explanation: This example command receives multiple lines of output for a transaction, TRAN1234. The output shows there are 1524 messages for the transaction that have an affinity to execute on IMS1. A transaction might have an affinity if it were an APPC, OTMA, or serial transaction. If an output line shows no affinity, like the example output line with a GBLQCNT of 14, then there could be a single IMS system indicated, or multiple IMS systems that have messages that contribute to the GBLQCNT.

Example for /DISPLAY UOR Command

Entry ET:

```
/DISPLAY UOR
```

Response ET:

```
ST  P-TOKEN  PSBNAME  RRS-URID                           IMS-TOKEN
 A           DBOVLFPC AF3499A27EC3E2980000000401010000 SYS1    0000000400000002
 EID=0001812223768AC1009B5870268346
 C           DBOVLFPB AF1124A27EE1C2380000000501020000 SYS1    0000000700000002
 EID=0002A35549021DA108538121766899670
RI  00010120 PLAPJK02 12345678901234567890123456789012 SYS1    0000001300000001
 EID=0001C35549021DB176523121445797320
*96337/145345*
```

Entry ET:

```
/DISPLAY UOR ACTIVE
```

Response ET:

```
ST  P-TOKEN  PSBNAME  RRS-URID                           IMS-TOKEN
 A           PLAPJK01 AF3278A27EC3E2980000000402010000 SYS1    0000000400000002
 LUWID=0EIMSNET.L62IMS1999E359820810001
*96338/091642*
```

Entry ET:

```
/DISPLAY UOR INDOUBT
```

Response ET:

```
ST  P-TOKEN  PSBNAME  RRS-URID                           IMS-TOKEN
RI  00010040 DBOVLFPC 98768883421097867890123456789012 SYS1    0000008300000001
 LUWID=0EIMSNET.L62IMS1763E358330410001
RI  00010120 PLAPJK02 AF34A6307EC4E2980000000201010000 SYS1    0000002700000001
 LUWID=0EIMSNET.L62IMS1438E339103810001
*96336/113502*
```

Entry ET:

```
/DISPLAY UOR 010140
```

Response ET:

```
ST  P-TOKEN  PSBNAME  RRS-URID                          IMS-TOKEN
RI  00010140 PLAPJK02 AF34A6307EC4E2980000000201010000 SYS1    0000002700000001
 LUWID=0EIMSNET.L62IMS1999E359820810001
*96340/011544*
```

Entry ET:

```
/DIS UOR
```

Response ET:

```
ST   P-TOKEN  PSBNAME RRS-URID                           IMS-RECTOKN
A             STLDDLT4 B6626FF77EB820000000000101010000 SYS3    00000001000
XID=RRMS000000220000002700000000000000000000000000000000000000000000000B662
6FF7574797C0D9D9D4E24BB6626FF7574797C000000000000000000000000000000000000
0000000000
*01246/150749*
```

Examples for /DISPLAY USER Command

The following are examples of the /DISPLAY USER command.

Example 1 for /DISPLAY USER Command

Entry ET:

```
/DISPLAY USER IMSUS01 IMSUS02
```

Response ET:

```
USER     USERID   ENQCT DEQCT   QCT
IMSUS01  IMSUS01      1     1     0 ALLOC(DTSLU201) CONV-HLD
N/A      IMSUS01    N/A   N/A   N/A ALLOC(L3270A  ) STATIC
N/A      IMSUS02    N/A   N/A   N/A ALLOC(L3270B  ) STATIC
DT327002 IMSUS02      0     0     0 ALLOC(DT327002)
*92350/112229*
```

Entry ET:

```
 /DISPLAY USER USER1 USER2 USER3 QCNT
```

Response ET:

```
 USERID   USER         GBLQCT
 N/A      USER1            25
 USER2    USER2A            0
 USER2    USER2B           10
 USER3    IMSUSR3A         40
 *95200/170817*
```

Entry ET:

```
/DISPLAY USER IMSUS01*
```

Response ET:

```
USER     USERID   ENQCT DEQCT   QCT
IMSUS01C N/A          0     0     0 STOPPED
IMSUS01D N/A          0     0     0 STOPPED
IMSUS01  IMSUS01      1     1     0 ALLOC(DTSLU201) CONV-HLD
N/A      IMSUS01    N/A   N/A   N/A ALLOC(L3270A  ) STATIC
*92350/113904*
```

Entry ET:

```
/DISPLAY USER ALL
```

Response ET:

```
USER     USERID   ENQCT DEQCT   QCT
CA13     N/A          0     0     0 STATIC
IMSUS01C N/A          0     0     0 STOPPED
IMSUS01D N/A          0     0     0 STOPPED
LU6SPA   N/A          0     0     0 ALLOC(LU6NDPF ) STATIC
LU6SPB   N/A          0     0     0 STATIC
DT327002 IMSUS02      0     0     0 ALLOC(DT327002)
IMSUS03  IMSUS03      0     0     0 ALLOC(DTSLU601)
IMSUS01  IMSUS01      1     1     0 ALLOC(DTSLU201) CONV-HLD
IMSUS05  N/A          0     0     0 STOPPED
N/A      IMSUS02    N/A   N/A   N/A ALLOC(L3270B  ) STATIC
N/A      IMSUS01    N/A   N/A   N/A ALLOC(L3270A  ) STATIC
N/A      IMSUS04    N/A   N/A   N/A ALLOC(L3270C  ) STATIC
*92350/114845*
```

Example 2 for /DISPLAY USER Command

Entry ET:

```
/DISPLAY USER ABC100 EMHQ QCNT
```

One user is on the queue.

Response ET:

```
14.42.46           57/DIS USER ABC100 QCNT EMHQ.
14.42.46           IEE600I REPLY TO 57 IS;/DIS USER ABC100 QCNT EMHQ.
14.42.46 JOB00110  DFS000I     USER          GBLQCT     IMS1
14.42.46 JOB00110  DFS000I     ABC100             0     IMS1
14.42.46 JOB00110  DFS000I     *98203/144246*   IMS1
```

Example 3 for /DISPLAY USER Command

Entry ET:

```
/DISPLAY USER IMSUS01 IMSUS02 IMSUS03 AUTOLOGON.
```

Response ET:

```
USER     NODE     MODE     DESC     ID
IMSUS01  N/A      N/A      N/A      N/A
IMSUS02  N/A      N/A      N/A      N/A
IMSUS03  N/A      N/A      N/A      N/A
*99243/082505*
```

Entry ET:

```
/CHANGE USER IMSUS01 AUTOLOGON DT327001 MODE LU032NT4.
```

Response ET:

```
DFS058I CHANGE COMMAND COMPLETED
```

Entry ET:

```
/CHANGE USER IMSUS02 AUTOLOGON DTSLU601 MODE LU6NEGPS ID IMSID2.
```

Response ET:

```
DFS058I CHANGE COMMAND COMPLETED
```

Entry ET:

```
/CHANGE USER IMSUS03 AUTOLOGON DT327001 LOGOND DFS3270 MODE LU032NT4.
```

Response ET:

```
DFS058I CHANGE COMMAND COMPLETED
```

Entry ET:

```
/DISPLAY USER IMSUS01 IMSUS02 IMSUS03 AUTOLOGON.
```

Response ET:

```
USER     NODE     MODE     DESC     ID
IMSUS01  DT327001 LU032NT4 N/A      N/A
IMSUS02  DTSLU601 LU6NEGPS N/A      IMSID2
IMSUS03  DT327001 LU032NT4 DFS3270  N/A
*99243/083309*
```

Example 4 for /DISPLAY USER RECOVERY Command

Entry ET:

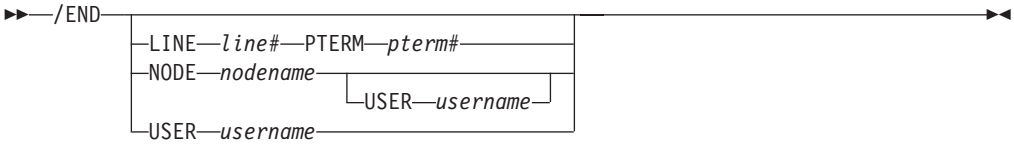
```
/DISPLAY USER IMSUS* IMSUS03 LU6SPG RECOVERY
```

Response ET:

```
USER OWNER SRM CONV STSN FPATH
IMSUS05 IMSA GLOBAL Y Y Y
IMSUS01 IMSA LOCAL Y Y Y
IMSUS03 IMSB GLOBAL Y N N
LU6SPG IMSA LOCAL Y Y Y
*99240/134730*
```

Chapter 22. /END

Format



Environments and Keywords

Table 56 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 56. Valid Environments for the /END Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/END	X		X
LINE	X		X
NODE	X		X
PTERM	X		X
USER	X		X

Usage

/END terminates all special operating modes established through the prior entry of an /EXCLUSIVE, /TEST, or /LOOPTEST command. /END NODE|USER does not reset TEST mode. Only the same terminal that is in TEST mode can reset the TEST mode (using /END without the NODE or USER keywords).

/END NODE applies to dynamic nodes in addition to static nodes because MFSTEST mode is associated with dynamic nodes as well as dynamic users. /END NODE and /END NODE USER reset MFSTEST mode at the node level./END USER resets MFSTEST at the user level. /END with no keywords resets MFSTEST at the node level for static terminals (they have no user level) and at the user level for dynamic terminals. The /END NODE USER command supports static and dynamic ISC sessions. When/END NODE or /END NODE USER commands are used for a dynamic node, only MFSTEST is reset. Exclusive mode and LOOPTEST mode can only be reset for a dynamic resource with the /END USER command. If global resource information is kept in Resource Manager, MFSTEST mode is reset globally and locally. If global resource information is not kept in Resource Manager, MFSTEST mode is reset locally.

/END with no keywords operates on the entering terminal with a signed on user. /END with no keywords is not supported through the OM API in an IMSplex.

LINE
Specifies the communication line for which special operating modes are terminated.

NODE
Specifies the VTAM node for which special operating modes are terminated.

© Copyright IBM Corp. 1974, 2003

/END NODE is not valid for ISC nodes with users or nodes that were dynamically created. /END NODE USER only applies to ISC sessions and affects the half-session allocated to username.

USER
Without the NODE keyword, USER specifies the dynamic user for which special operating modes are terminated. The named user must exist in IMS, must be dynamic, and must not be signed on to a node. When /END USER is issued against a temporary user, which was created solely to retain status that is now reset, the temporary user is deleted at the next simple checkpoint.

Example for /END Command

Entry ET:

```
/END
```

Response ET:

```
DFS058I  END COMMAND COMPLETED
```

Explanation: All modes that previously existed are ended.

Chapter 23. /ERESTART

Format

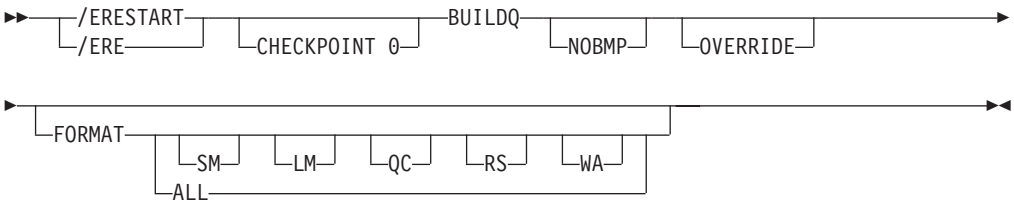
Manual Restart of an XRF Alternate System

Use this command only after the active system issues message DFS3804I.

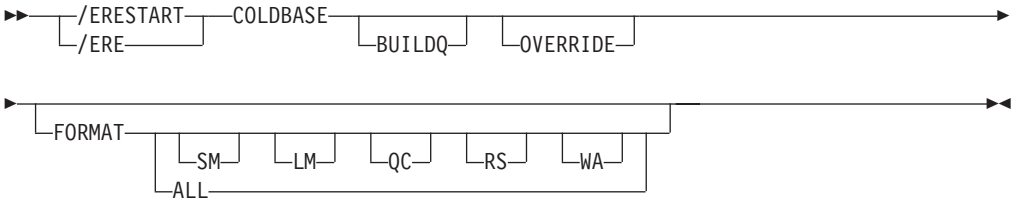
Restart of IMS Following Loss of Virtual Storage Only

Restart of IMS Following Loss of Virtual Storage and Message Queue Data Set Integrity

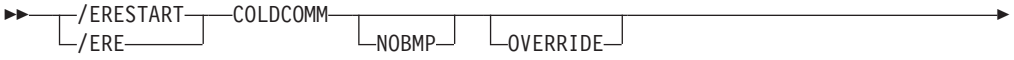
The message queues have not been dumped to the system log since the most recent cold start.



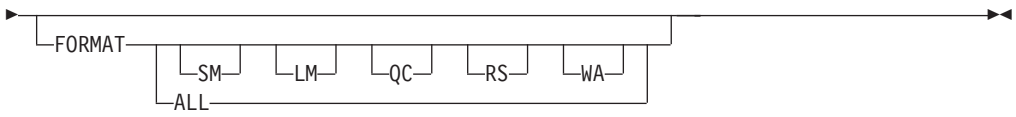
Restart of IMS Following /ERESTART Failure of the Database Component



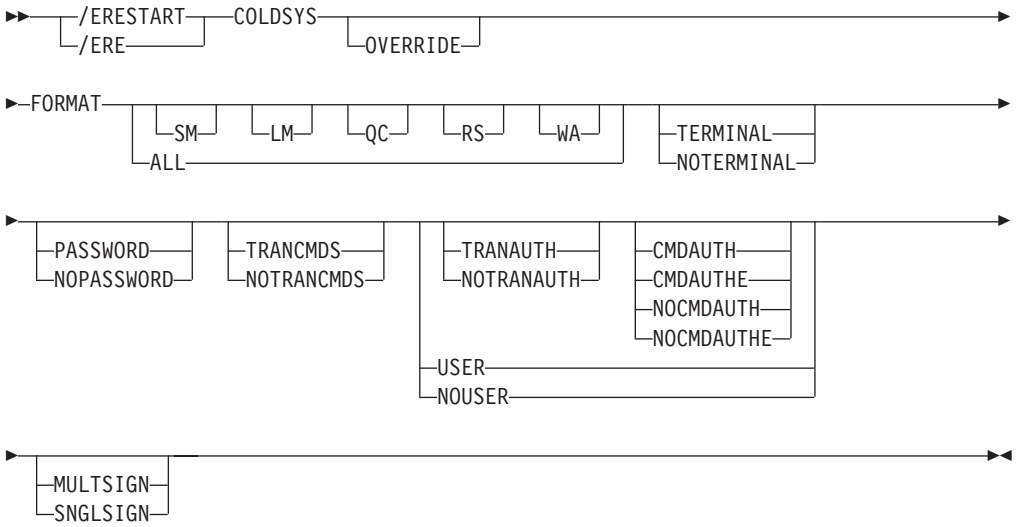
Restart of IMS Following /ERESTART Failure of Communication Component



© Copyright IBM Corp. 1974, 2003



Restart of IMS Following /ERESTART Failure of Both the Database and Communication Components



Environments and Keywords

Table 57 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 57. Valid Environments for the /ERESTART Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/ERESTART	X	X	X
BACKUP	X		X
BUILDQ	X		X
CHECKPOINT	X	X	X
CMDAUTH	X		X
CMDAUTHE	X		X
COLDBASE	X	X	
COLDCOMM	X		X
COLDSYS	X	X	X
FORMAT	X	X	X
MULTSIGN	X		X
NOBMP	X	X	X
NOCMDAUTH	X		X
NOCMDAUTHE	X		X
NOPASSWORD	X		X

Table 57. Valid Environments for the /ERESTART Command and Keywords (continued)

Command / Keywords	DB/DC	DBCTL	DCCTL
NOTERMINAL	X		X
NOTRANAUTH	X		X
NOTRANCMDS	X		X
NOUSER	X		X
OVERRIDE	X	X	X
PASSWORD	X		X
SNGLSIGN	X		X
TERMINAL	X		X
TRANAUTH	X		X
TRANCMDS	X		X
USER	X		X

Usage

/ERESTART is a multisegment command used to:

- Perform a manual restart of an IMS XRF alternate system.
- Restart IMS any time the system was not terminated with an orderly /CHECKPOINT shutdown. Three conditions that result in the need for an emergency restart are:
  - Abnormal termination of IMS
  - Abnormal termination of z/OS
  - Forced termination of IMS using the z/OS MODIFY command

For an /ERESTART command restart, the base security definition is created:

- From the IMS system definition and the EXECUTE parameter specifications, when the COLDSYS keyword is specified.
- From the IMS checkpoint data, when the COLDSYS keyword is not specified.

To override the security definitions, specify the COLDSYS keyword and the security keywords on the COLDSYS keyword.

The format for /ERESTART depends on whether or not the IMS system is an XRF alternate system. If the system is not an alternate system, the format also depends on the type of system failure associated with the termination and whether or not the message queues have been dumped to the system log since the most recent cold start.

All /ERESTART formats require an EOM indication to denote end-of-message; an EOS indication must be included for all segments that precede the last segment. See “Multisegment Command Input” on page 8 for more details on using EOS and EOM.

With Fast Path, the /ERESTART command loads the latest MSDB checkpoint data set and applies all MSDB changes up to the latest complete commit point. /ERESTART also ensures that DEDB updates created between the restart checkpoint and the latest complete commit point are applied. If the DEDB updates are not applied, they are queued for output. Output messages not acknowledged or not transmitted are also queued for output.

When IMS initializes, the system parameters used for this initialization can come from the IMS system generation, from a PROCLIB member, or from EXEC statements that can override both the defaults and the PROCLIB members. Therefore, message DFS1929I is displayed showing the system parameters used for this particular initialization. The system parameters are also written to the job log.

For a list of the commands recovered, see "Commands Recovered During Emergency Restart" on page 26

BACKUP
Indicates that the control region is an alternate system in an XRF environment. This command is required when performing manual restart.

BUILDQ
Is required for any restart from a failure in which message queue data set integrity was lost. When a checkpoint other than checkpoint 0 is specified, the checkpoint must be one at which the message queues were dumped to the system log. Message queue data set integrity has been lost if one or more of the data sets have been reallocated or reformatted. If a checkpoint is not specified, IMS selects the appropriate checkpoint from which to start.

If an /ERESTART BUILDQ command fails, and then you issue the /ERESTART CHECKPOINT 0, /ERESTART COLDCOMM, or /ERESTART COLDSYS command, messages in local queues are lost. However, the IMS Message Requeuer (MRQ) program product (5655-038) can be used to recover local message queues. For more information about MRQ, refer to the *IMS/ESA Message Requeuer Program Description/Operations Manual*.

In a shared-queues environment, the BUILDQ keyword is ignored because the message queue data sets are not used.

CHECKPOINT
CHECKPOINT is an optional keyword that identifies whether the restart should use the initial system checkpoint taken during the cold start of the IMS subsystem. If you do not specify this keyword, IMS determines the correct checkpoint to use. If you do specify this keyword, you can only specify CHECKPOINT 0 to use the initial system checkpoint.

CMDAUTH
Specifies that both signon (user identification verification) and command authorization for static and ETO terminals are in effect at the end of the emergency restart. (Command authorization is same as specifying RCF=S on the startup parameter.)

To specify CMDAUTH, either:

```
SECURITY macro TYPE=RACFTERM|RACFCOM|SIGNEXIT|TRANEXIT
```

must be specified in the IMS system definition, or

```
EXEC parameters RCF=A|Y|T|C|S
```

must be specified on an EXEC parameter.

CMDAUTHE
Specifies that command authorization for ETO terminals (same as RCF=S on the startup parameter) is in effect at the end of the emergency restart. CMDAUTHE also resets command authorization for static terminals, if it was set.

To specify CMDAUTHE, either:

```
SECURITY macro TYPE=RACFTERM|RACFCOM|SIGNEXIT|TRANEXIT
```

must be specified in the IMS system definition, or

```
EXEC parameters RCF=A|Y|T|C|S
```

must be specified on an EXEC parameter.

COLDBASE

Indicates a cold start of the database component, while performing an emergency restart of the communications component.

- If this keyword is used, the recovery of the databases is the responsibility of the user. The Fast Path areas will not be redone and no backouts of inflight DL/I databases will be performed. If in-doubts exist, a batch backout run with the cold start option will backout inflight DL/I data. This will place both DL/I and Fast Path data in the aborted state.
- If this keyword is not used, the database component will be warm started.

COLDCOMM

Indicates a cold start of the data communication component, while an emergency restart of the database component is being performed. This includes recovering Fast Path DEDBs, reloading MSDBs, backing out inflight changes to DL/I databases, and maintaining all existing indoubt data. COLDCOMM is used to get the DC network reinstated as soon as possible while databases are being recovered. As a result of COLDCOMM, all of the messages on the local message queue will be lost; inflight messages in shared queues are moved to the cold queue for later recovery. If this keyword is not used, the data communication component will be restarted.

COLDSYS

Indicates a cold start of both the database and the data communication components. The `/ERESTART COLDSYS` command performs the same function as the `/NRESTART CHECKPOINT 0 DETACH` command performed in prior releases, and replaces it.

The SGN=, TRN=, and RCF= startup parameters can be overridden by the `/ERESTART COLDSYS` command using the keywords shown in Table 58 Table 58 lists the keywords and the startup parameters that those keywords can override and a brief description.

Table 58. Security Keywords and Their Startup Parameter Equivalents

Keyword[1]	Description	Startup Parameter
CMDAUTH	RACF command authorization on static and ETO terminals only	RCF=S
CMDAUTHE	RACF command authorization on ETO terminals only	RCF=C
MULTSIGN	Permits multiple signons for each user ID	SGN=M
NOCMDAUTH	Resets the command authorization on static and ETO terminals	Not RCF=S
NOCMDAUTHE	Resets the command authorization on ETO terminals only	Not RCF=C
NOTRANAUTH	Resets the transaction authorization.	Not TRN=F or Y
NOUSER	Resets user identification verification, transaction authorization, and command authorization	Not SGN=F or Y (G or Z becomes M) Not TRN=F or Y Not RCF=C or S

Table 58. Security Keywords and Their Startup Parameter Equivalents (continued)

Keyword[1]	Description	Startup Parameter
SNGLSIGN	Permits a single signon for each user ID	SGN=F and Y Not SGN=M (G or Z becomes F or Y)
TRANAUTH	Transaction authorization	TRN=F or Y
USER	Sets user identification verification	SGN=Y

Note:

1. Valid only with the /ERESTART COLDSYS command.

FORMAT

Specifies which queues or data sets should be formatted as part of the restart process when:

- A message queue or data set I/O error occurs.
- The size of a message queue or data set is to be changed.
- A message queue or data set is to be reallocated.
- Allocation of the WADS changes.

When queues or data sets must be formatted as part of the restart process, specify one or more of the following:

SM Short-message queue

LM Long-message queue

QC Control record data set (QBLKs)

RS Restart data set

WA Write ahead data set

MD MSDB dump data set (valid only for backup)

ALL All message queues (SM and LM) and data sets (QC, WA and RS). For DBCTL, this supports RS and WA only.

Any combination of SM, LM, QC, WA, MD, and RS can be specified; for example, FORMAT LM RS. When you specify ALL, do not specify SM, LM, QC, WA, MD, or RS.

In a shared-queues environment, the LM, SM, and QC parameters are ignored because the message queue data sets are not used. If you specify ALL, IMS does not attempt to format the message queue data sets.

Table 59 shows the environments in which the parameters are valid.

Table 59. /ERESTART FORMAT Command Parameter Environments

Parameter	DB/DC	DBCTL	DCCTL
SM	X		X
LM	X		X
QC	X		X
RS	X	X	X
WA	X	X	X
MD	X		X

Table 59. /ERESTART FORMAT Command Parameter Environments (continued)

Parameter	DB/DC	DBCTL	DCCTL
ALL	X	X	X

If the WADS must be reformatted during emergency restart, the ALL parameter can be used. IMS first closes the OLDS from the WADS and then reformats the WADS. If you generally use the ALL parameter during restart, do not be concerned that the WADS will not be available to close the OLDS.

MULTSIGN
Permits multiple signons for each user ID.

In an IMSplex with Resource Manager and a resource structure, if MULTSIGN conflicts with the single user signon definition for the IMSplex, a warning message will be issued.

NOBMP
Specifies no backout of BMP updates occurs and all affected databases and programs are stopped.

If NOBMP is not specified, all updates made subsequent to the last commit point invoked by the active BMP programs are backed out of the database as part of the restart process.

NOBMP is ignored if no BMP programs were active at the time of the system failure.

NOCMDAUTH
Resets command authorization on static and ETO terminals.

NOCMDAUTHE
Resets command authorization for static and ETO terminals. The command authorization is reset for static terminals because the command authorization for static terminals cannot exist without the command authorization for ETO terminals.

NOTERMINAL
Specifies that the terminal security specifications established by the Security Maintenance utility are not in effect at completion of this emergency restart.

IMS system definition might have precluded the authority of the master terminal operator to negate terminal security, in which case, if NOTERMINAL is specified, an error message is received.

NOTRANAUTH
Turns off transaction authorization. NOTRANAUTH is not the opposite of TRANAUTH. TRANAUTH sets transaction authorization and also turns on signon (user identification verification).

If you specify NOTRANAUTH, it will be rejected with an error message if either:

- SECLEVL=FORCTRAN was specified on the system definition SECURITY macro.
- TRN=F was specified as a JCL EXEC parameter.

NOUSER
Specifies that none of the following is in effect at the end of the emergency restart:

- Transaction authorization
- User identification verification

- Command authorization

OVERRIDE
Is required only to restart the system after failure of power, machine, z/OS, or DBRC where IMS abnormal termination was unable to mark the DBRC subsystem record in RECON as abnormally terminated. IMS emergency restart will abort with message DFS0618A when DBRC indicates that the subsystem is currently active and that neither the OVERRIDE keyword nor the BACKUP keyword is present on the /ERESTART command. If there is any doubt about the status of an IMS system at restart time, the OVERRIDE keyword should not be used.

Attention: Use of the OVERRIDE keyword on a currently running IMS system can lead to database and system integrity problems.

PASSWORD or **NOPASSWORD**
Specifies whether (PASSWORD) or not (NOPASSWORD) the password security specifications established by the Security Maintenance utility will be in effect at completion of this emergency restart.

IMS system definition might have precluded the authority of the master terminal operator to negate password security, in which case, if NOPASSWORD is specified, an error message is received.

SNGLSIGN
Permits a single signon for each user ID.

In an IMSplex with Resource Manager and a resource structure, if SNGLSIGN conflicts with the single user signon definition for the IMSplex, a warning message is issued.

TERMINAL
Specifies that the terminal security specifications established by the Security Maintenance utility are in effect at completion of this emergency restart. However, TERMINAL can only be used for transaction authorization if RCF=S is specified. SMU is not used for command terminal security if RACF is used.

TRANAUTH
Specifies both transaction authorization and user identification verification, with or without RACF.

To specify TRANAUTH either:

```
SECURITY macro TYPE=RACFTERM|RACFCOM|SIGNEXIT|TRANEXIT
```

must be specified in the IMS system definition, or

```
EXEC parameters RCF=A|Y|T|C|S
```

must be specified on an EXEC parameter.

TRANCMDS or **NOTRANCMDS**
Specifies whether (TRANCMDS) or not (NOTRANCMDS) the transaction command security established by the Security Maintenance utility is in effect at completion of this emergency restart.

TRANCMDS
Specifies that transaction command security established by the Security Maintenance utility is in effect at completion of this emergency restart.

NOTRANCMDS
Specifies that transaction-entered commands cannot be entered. IMS system definition might have precluded the authority of the master terminal

operator to not allow transaction-entered commands, in which case, if the NOTRANCMDS keyword is used, an error message is returned.

USER

Specifies user identification verification. User identification verification means that signon is required by the static terminals that are defined to SMU for signon verification. This keyword has no effect on ETO terminals because they are always required to sign on. User identification verification can be forced on by TRANAUTH or CMDAUTH.

To specify USER, either:

```
SECURITY macro TYPE=RACFTERM|RACFCOM|SIGNEXIT|TRANEXIT
```

must be specified in the IMS system definition, or

```
EXEC parameters RCF=A|Y|T|C|S
```

must be specified on an EXEC parameter.

Examples

Example 1 for /ERESTART Command

All system data sets are intact.

Entry ET:

```
/ERESTART.
```

Response ET:

```
DFS058I  (time stamp) ERESTART COMMAND IN PROGRESS
DFS680I USING CHKPT 85200/123456
DFS994I *CHKPT 85201/110117**SIMPLE*
```

Explanation: The restart is being performed from checkpoint 85200/123456, which was selected by IMS. If IMS contained active BMPs when the system failed, checkpoint 85200/123456 is the checkpoint that allows backout of all updates made by each active BMP since its latest system checkpoint. A simple checkpoint is written on the system log. The checkpoint number is 85201/110117.

Example 2 for /ERESTART Command

Message queues are in error.

Entry ET:

```
/ERESTART BUILDQ.
```

Response ET:

```
DFS058I  (time stamp) ERESTART COMMAND IN PROGRESS
DFS680I  USING CHKPT 85119/074811
DFS994I  *CHKPT *85120/192021**SIMPLE*
```

Explanation: IMS is restarted at 192021 (time) on 85120 (* * date) from the most recent DUMPQ or SNAPQ checkpoint, 85119/074811. A simple checkpoint is written on the system log. The checkpoint number is 85120/192021.

The message queues from checkpoint 85119/074811 are loaded.

If BMP programs were active when the system failed, all updates invoked by the BMP programs are backed out.

Example 3 for /ERESTART Command

Small and large message queues were reallocated. The message queues have been dumped at least once since the most recent cold start.

Entry ET:

```
/ERESTART BUILDQ FORMAT SM LM.
```

Response ET:

```
DFS058I  (time stamp) ERESTART COMMAND IN PROGRESS
DFS680I  USING CHKPT 85201/070348
DFS994I  *CHKPT *85204/010203**SIMPLE*
```

Explanation: IMS is restarted at 010203 (time) on 85204 (Julian date). A simple checkpoint was written on the system log. The checkpoint number is 85204/010203.

The large and small message queue data sets are reformatted.

The message queues are loaded from checkpoint 85201/070348.

If BMP programs were active when the system failed, all updates invoked by the BMP programs are backed out.

Example 4 for /ERESTART Command

Message queues in error or data sets reallocated and the message queues have not been dumped since the most recent cold start.

Entry ET:

```
/ERESTART BUILDQ FORMAT ALL NOBMP.
```

Response ET:

```
DFS058I  (time stamp) ERESTART COMMAND IN PROGRESS
DFS680I  USING CHKPT 85045/112140
DFS994I  *CHKPT 85121/235959**SIMPLE*
```

Explanation: IMS is restarted at 235959 (time) on 85121 (Julian date) from the checkpoint taken during cold start 85045/112140. A simple checkpoint is written on the system log. The checkpoint number is 85121/235959.

All message queues and data sets are reformatted.

The message queues from the system log were loaded.

BMP programs that were active when the system failed are stopped, as are the databases updated by them.

Example 5 for /ERESTART Command

Figure 4 on page 363 and Figure 5 on page 363 show formatted master screens for active and backup IMS master terminals in an XRF environment.

The `/ERESTART BACKUP` command was entered on the master terminal of the XRF IMS system IMSB. During the synchronization phase, the alternate system

requested a SNAPQ checkpoint from the active system. The /ERESTART BACKUP is currently complete and alternate system IMSB is now tracking active system IMSA.

```
02/05/15 14:28:44  RSENAME: DFSRSENM  BACKUP  TRACKING IN PROGRESS  IMSB
DFS810A IMS READY 085135/1422535 IMS432  .IEFPROC
 DFS058 14:26:10 ERESTART COMMAND IN PROGRESS
DFS3871I JOINAVM WAS SUCCESSFUL
DFS3873I JOINRSE MODE=BACKUP WAS SUCCESSFUL
DFS3857I OKTOSYNC ARRIVED FROM THE ACTIVE SYSTEM.
DFS680I USING CHKPT85135/142629
 DFS2591I NO MSDB HEADERS FOUND, IMAGE COPY LOAD IGNORED
DFS3839I 14:26:46 XRF INITIAL DC PRE-OPEN COMPLETE.
DFS3838I 14:28:41 XRF INITIAL DB PRE-OPEN COMPLETE.
--------------------------------------------------------------------------------









                                                      PASSWORD:
/ere backup.
```

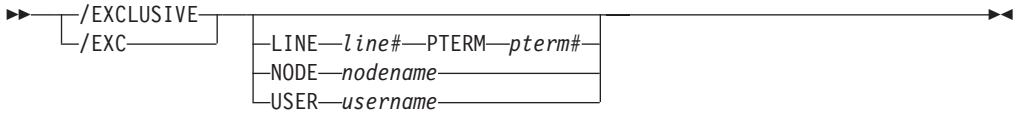
Figure 4. IMS Formatted Master for the Alternate System

```
02/05/15 14:26:40  RSENAME: DFSRSENM  ACTIVE                            IMSA
DFS3499I ACTIVE DDNAMES: MODBLKSA IMSACBA  FORMATA  MODSTAT ID:   11
DFS3804I LAST CHKPT ID VALID FOR RESTART: 85135/141619-BUILDQ: 85135/141619

DFS994I COLD START COMPLETED.
 DFS3830 14:16:23 NO DEFERRED PAGE FIXING WAS DONE
DFS3856I REQSYNC WAS PROCESSED TO TAKE SNAPQ CHKPT.
DFS2716I NO MSDBS FOUND - NO MSDB CHECKPOINT TAKEN
 DFS994I *CHKPT 85135/14269***SNAPQ**
DFS3499I ACTIVE DDNAMES: MODBLKSA IMSACBA  FORMATA MODSTAT ID:   11
--------------------------------------------------------------------------------









                          MASTER MESSAGE WAITING    PASSWORD:
```

Figure 5. IMS Formatted Master for the Active System

Chapter 24. /EXCLUSIVE

Format



Environments and Keywords

Table 60 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 60. Valid Environments for the /EXCLUSIVE Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/EXCLUSIVE	X		X
LINE	X		X
NODE	X		X
PTERM	X		X
USER	X		X

Usage

/EXCLUSIVE puts a terminal into exclusive mode. This command, without keywords, is valid from a terminal with a signed on user. The LINE PTERM and NODE keywords are only valid for output-only terminals. For a dynamically created user, the exclusive mode status is remembered across signons.

Exclusive mode restricts the output received by the terminal affected:

- A regular input/output terminal in exclusive mode only receives messages transmitted in response to transactions entered from that physical terminal.
- An output-only terminal in exclusive mode receives messages transmitted in response to transactions entered into its associated input logical terminals.

Except for an IMS system message responding to an error condition, all other output for a terminal in exclusive mode remains queued for later transmission. The queued output is transmitted after exclusive mode is reset with the /END command or a /START command.

In an IMSplex, if global resource information is kept in Resource Manager, the /EXCLUSIVE command sets a global exclusive status for the resource. If global resource information is not kept in Resource Manager, the /EXCLUSIVE command sets the status.

/END is used to get a terminal out of exclusive mode.

LINE PTERM
Specifies the communication line to be put into exclusive mode. The /DISPLAY LINE line# PTERM pterm# command identifies a terminal in exclusive mode.

© Copyright IBM Corp. 1974, 2003

NODE
Specifies that the static VTAM node be put into exclusive mode. This command is not valid for nodes that were dynamically created.

USER
Specifies that the dynamic user be put into exclusive mode. If the user does not exist in IMS, it and its associated LTERMs will be created to remember the exclusive status. If `/EXCLUSIVE USER` is issued against an existing user, the user must be dynamic.

Examples

Example 1 for /EXCLUSIVE Command

Entry ET:

```
/EXCLUSIVE
```

Response ET:

```
DFS058I  EXCLUSIVE COMMAND COMPLETED
```

Explanation: No messages are sent to this terminal other than application program output resulting directly from transaction input.

Example 2 for /EXCLUSIVE Command

Entry ET:

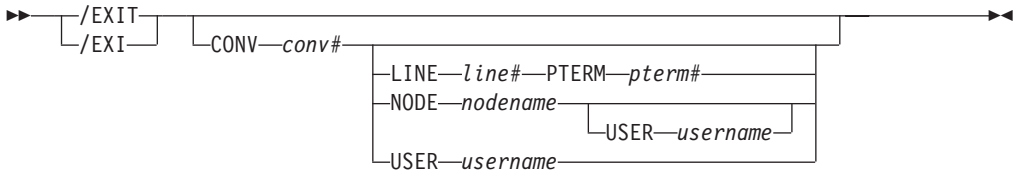
```
/EXCLUSIVE USER alpha
```

Response ET:

```
DFS058I  EXCLUSIVE COMMAND COMPLETED
```

Chapter 25. /EXIT

Format



Environments and Keywords

Table 61 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 61. Valid Environments for the /EXIT Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/EXIT	X		X
CONVERSATION	X		X
LINE	X		X
NODE	X		X
PTERM	X		X
USER	X		X

Usage

The /EXIT command ends conversations between users and application programs. The conversation can be either active or in hold status at the time /EXIT is entered. Depending upon the conversational processing options selected for the IMS system, the application program might or might not be notified of the conversation termination.

Transactions that are queued for processing when the /EXIT command is issued are discarded. If a transaction is being processed when /EXIT is issued, the command is rejected.

The /EXIT command with no keywords can only be used if the conversation is active.

If global resource information is kept in Resource Manager (RM), /EXIT NODE or /EXIT USER terminates the conversation in RM. If global resource information is not kept in RM, /EXIT NODE or /EXIT USER terminates a conversation on the IMS where the /EXIT command is entered or routed to by the OM SPOC.

/EXIT is not valid from an LU 6.2 device or OTMA device. DEALLOCATE the APPC conversation after receiving all output from an iteration. When the /EXIT command specifies only the CONVERSATION keyword, the command can be entered only from the terminal that owns the conversation.

© Copyright IBM Corp. 1974, 2003

CONVERSATION
Terminates the specified conversation. The conversation is terminated whether it was active or held. The conversation identifier (conv#) must be specified as a 4-digit number, including leading zeros.

LINE PTERM
Terminates the conversation on the physical terminal specified, regardless of whether the conversation was active or held. The line specified must be in stopped and idle status.

NODE
Terminates the conversation on the node specified, regardless of whether the conversation was active or held. The node specified must be stopped or idle. This form of the command does not support dynamic nodes or LU 6.2 nodes defined with users.

The NODE USER combination applies only to the specified half-session of the ISC parallel session node allocated to USER The half-session must have been stopped and idled. The command must be issued from a terminal or node other than the one in the conversation.

USER
Terminates the conversation associated with the specified user. The user must not be signed on to a node. The user must be dynamic.

Examples

Example 1 for /EXIT Command

Entry ET:

```
/EXIT
```

Response ET:

```
DFS058I  EXIT COMMAND COMPLETED
```

Explanation: The active and not held conversation associated with the entering terminal is terminated.

Example 2 for /EXIT Command

Entry ET:

```
/EXIT
```

Response ET:

```
DFS576I  EXIT COMPLETED, TRANSACTION DISCARDED
```

Explanation: /EXIT processing found the conversational transaction enqueued for an application or transmission across an MSC link, but it has not yet been scheduled for processing. The transaction has been discarded.

Example 3 for /EXIT Command

Entry ET:

```
/EXIT
```

Response ET:

```
DFS577I  EXIT COMPLETED, TRANSACTION STILL ACTIVE
```

Explanation: /EXIT processing found the conversational transaction had been or is in the process of being transmitted across an MSC link.

Example 4 for /EXIT Command

Entry ET:

```
/EXIT CONVERSATION 0001
```

Response ET:

```
DFS058I  EXIT COMMAND COMPLETED
```

Explanation: The active or held conversation (0001) is terminated if the terminal is in conversation.

Example 5 for /EXIT Command

Entry ET:

```
/EXIT CONVERSATION 0001 LINE 10 PTERM 2
```

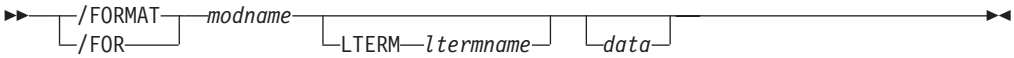
Response ET:

```
DFS058I  EXIT COMMAND COMPLETED
```

Explanation: Used from another PTERM or master terminal when a PTERM in conversation is “locked,” waiting for a response. /EXIT CONVERSATION should be used before a /START LINE command that resets all conversations active on the line.

Chapter 26. /FORMAT

Format



Environments and Keywords

Table 62 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keyword can be issued.

Table 62. Valid Environments for the /FORMAT Command and Keyword

Command / Keyword	DB/DC	DBCTL	DCCTL
/FORMAT	X		X
LTERM	X		X

Usage

/FORMAT causes a specific format to be displayed on a physical terminal using the IMS Message Format Service (MFS). The displayed format is used by IMS in processing the next input from the formatted terminal.

Using a password with /FORMAT is not valid.

/FORMAT, where an IMS-supplied default modname is used, is not effective for a 3270 master terminal that uses the MFS master terminal formatting option.

Recommendation: If a format is lost during a conversation, do **not** use the /FORMAT command to restore the format. Instead, enter a /HOLD command followed by a /RELEASE command to display the first physical page of the current message and restore the format.

modname
Specifies the name of the message output descriptor (MOD) to be used to format the terminal.

LTERM
Specifies the name of the logical terminal to be formatted. LTERM is required when a terminal other than the input terminal is to be formatted. If LTERM is not specified, the input physical terminal must have at least one LTERM assigned.

If the ltermname does not exist, an attempt is made to create the lterm and associated user structure. The command creates this structure by queueing the format and data, if any exists, to the terminal as if it were a message. This process is similar to a message switch, a message insert, or a /BROADCAST LTERM, all of which queue a message to an LTERM.

The format request is rejected if the specified LTERM is:

- Assigned to a physical terminal that is not supported by MFS
- In line-response mode or exclusive mode
- Involved in an active conversation
- An input-only terminal

© Copyright IBM Corp. 1974, 2003

- Not the alternate master in an XRF system
- A remote LTERM

In an IMSplex, /FORMAT LTERM can be used to queue a format to an LTERM that is not active or that is active on any IMS in the IMSplex.

data
Is a string consisting of 1 to 8 characters to be inserted into the output message created for the terminal being formatted. Since MFS treats the data as application program output data, the data string can be used to select a logical page (LPAGE) within the requested MOD. The way the selection is performed is determined by the requested MOD's LPAGE definitions.

Example for /FORMAT Command

Remote Terminal entry:

```
/FORMAT DFSMO4
/BROADCAST LTERM WTOR (eos)
this is segment 1 (eos)
this is segment 2 (eos)
this is segment 3 (eom)
```

Response ET:

```
 DFS058I  BROADCAST COMMAND COMPLETED
```

Response RT:

```
THIS IS SEGMENT 1
THIS IS SEGMENT 2
THIS IS SEGMENT 3
```

Explanation: The remote terminal is first formatted by the /FORMAT command, where default format DFSMO4 supports the input of 4 segments. This is followed by /BROADCAST with four segments.

Chapter 27. /HOLD

Format

Environments

Table 63 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command can be issued.

Table 63. Valid Environments for the /HOLD Command

Command	DB/DC	DBCTL	DCCTL
/HOLD	X		X

Usage

The /HOLD command suspends and saves a currently active conversation.

The IMS response to /HOLD provides a conversation identification that must be used later when the held conversation is resumed (/RELEASE command) or terminated (/EXIT).

In an IMSplex, if global resource information is kept in Resource Manager, /HOLD suspends and saves the active conversation globally in Resource Manager. If global resource information is not kept in Resource Manager, /HOLD suspends and saves the active conversation locally.

Example for /HOLD Command

Entry ET:

```
/HOLD
```

Response ET:

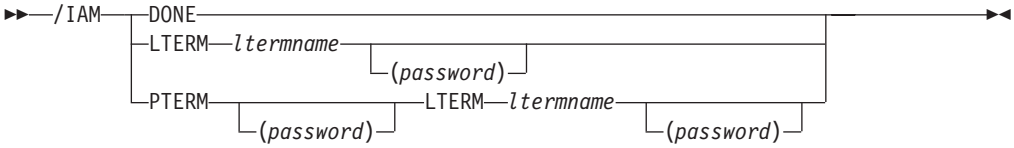
```
DFS999I  HELD CONVERSATION ID IS 0001
```

Explanation: The active conversation is saved and is assigned an identification of 0001.

© Copyright IBM Corp. 1974, 2003

Chapter 28. /IAM

Format



Environments and Keywords

Table 64 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 64. Valid Environments for the /IAM Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/IAM	X		X
DONE	X		X
LTERM	X		X
PTERM	X		X

Usage

/IAM is used to sign on to IMS from a terminal that is on a non-VTAM attached switched communication line. This command (without the DONE keyword) must be entered before any input transaction codes or terminal commands are accepted.

The terminal operator has five chances to sign on. If signon is not accomplished after five tries, IMS sends the operator a message indicating that the line is being disconnected. IMS disconnects the line and restores the line to answering status.

/IAM is not supported for Network Terminal Option devices.

DONE
Requests that IMS disconnect your terminal.

LTERM
Indicates this command automatically accomplishes the attachment of a user logical terminal named ltermname to the switched (dialup) communication line over which the call was received from the remote terminal.

If the first four characters of the ltermname parameter are INQU, the connection is completed with the inquiry logical terminal associated with the line/pterm that received the call.

PTERM [password] LTERM ltermname [password]
All logical terminals associated with the user in which logical terminal ltermname exists are signed on to the terminal entering the command. This form of /IAM is required if conversational transactions are to be entered.

© Copyright IBM Corp. 1974, 2003

Examples

Example 1 for /IAM Command

Entry ET:

```
/IAM LTERM SMITH
```

Response ET:

```
DFS058I  IAM COMMAND COMPLETED
```

Explanation: Logical terminal SMITH is signed on to IMS.

Example 2 for /IAM Command

Entry ET:

```
/IAM LTERM INQU
```

Response ET:

```
DFS058I  IAM COMMAND COMPLETED
```

Explanation: The inquiry logical terminal associated with the line/pterm that received the call is signed on to IMS.

Example 3 for /IAM Command

Entry ET:

```
/IAM PTERM (DOLLY) LTERM SUE (GIRL)
```

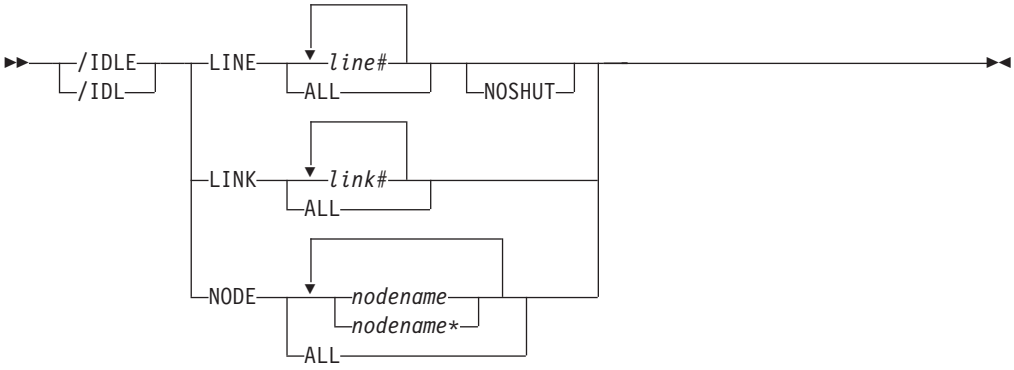
Response ET:

```
DFS058I  IAM COMMAND COMPLETED
```

Explanation: All logical terminals associated with the user to which LTERM SUE belongs are signed on to IMS.

Chapter 29. /IDLE

Format



Environments and Keywords

Table 65 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 65. Valid Environments for the /IDLE Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/IDLE	X		X
LINE	X		X
LINK	X		X
NODE	X		X
NOSHUT	X		X

Usage

`/IDLE` is used to immediately terminate input from, or output to, physical terminals assigned to the lines, logical links, or nodes specified in the command. Partially processed input messages on the lines specified in the command are discarded. Output messages being sent are returned to the message queues for later transmission.

LINE
Specifies the communication line to be immediately terminated.

`/IDLE` does not apply to the system console line, the master terminal line, or the secondary master terminal line. If these lines are specified, a `DFS058 COMMAND COMPLETED EXCEPT LINE X` message is returned. If the master terminal is assigned to a line that must be idle, the master terminal must be reassigned to a different line before `/IDLE` is specified.

NOSHUT
Immediately terminates input to and output from 3270 remote BTAM lines without a checkpoint shutdown. `/IDLE` cannot be entered before a checkpoint unless the NOSHUT keyword is included in the command. NOSHUT is only valid if IMS is not in the process of shutting down and is mutually exclusive with

© Copyright IBM Corp. 1974, 2003

the NODE parameter. When /IDLE LINE line# NOSHUT is entered, the 3270 remote BTAM lines must be stopped or process stopped. A DFS058 COMMAND COMPLETED EXCEPT LINE X message is returned if the command is entered and one of the following conditions exists:

- The line is the master terminal line or system console line, including the secondary master terminal line.
- The line is not stopped or process stopped.
- The line is not a 3270 remote BTAM line.
- The line is not open.
- The line is deactivated by IMS.

If the line specified with the NOSHUT keyword belongs to a line group, the input and output for all lines within the group are immediately terminated. If the line belonging to the line group has the master terminal assigned to it, the /IDLE LINE line# NOSHUT command is rejected.

To restart the lines idled by the /IDLE LINE line# NOSHUT command, either the /START LINE or /RSTART LINE command must be entered.

LINK

Specifies the logical link from which input, or to which output, is to be terminated. (No input messages are lost with /IDLE LINK.)

If a checkpoint shutdown is not in progress, only MSC VTAM links are valid. All non-MSC VTAM links require a shutdown checkpoint in progress. IMS returns a DFS058 IDLE COMMAND COMPLETED EXCEPT LINK X message if one of the following conditions exists:

- The link is idled already.
- The link is not connected.
- The link cannot be opened.
- A shutdown checkpoint is not in progress (for non-MSC VTAM links).

An /IDLE LINK link# command internally causes issuing of the VTAM command:

```
VARY NET,TERM,TYPE=FORCE,NOTIFY=NO,SCOPE=ALL,LU1=xxxxxxxx,LU2=yyyyyyyy
```

All logical links, assigned to the same physical link, associated with the named logical link (link#) will be stopped. That is, all parallel sessions on the same physical link will be stopped. The /RSTART LINK command is used to reactivate links or sessions.

The /IDLE LINK link# command should not be used as the first attempt to stop the link. If the MSC link appears to be hung, the IMS operator should:

1. First attempt to stop the link with the IMS /PSTOP command.
2. If this fails, display the link session status through VTAM, using the VTAM command:

   ```
   D NET,ID=xxxxxxxx
   ```

   or the VTAM command:

   ```
   D NET,SESSIONS,LIST=ALL
   ```
3. If the link session is still active (STATUS=ACTIV), terminate it with the /IDLE LINK command or the VTAM command:

   ```
   VARY NET,TERM,TYPE=FORCE,LU1=xxxxxxxx
   ```

Note: All parallel sessions associated with the link will be terminated.

NODE

Specifies the VTAM node to which output is to be terminated, or from which input is to be terminated. The master terminal only can be idled when specifically named.

If an `/IDLE NODE nodename` command is entered for an ISC node defined with users, all half-sessions of the specified node are affected.

An `/IDLE NODE ALL` command causes the VTAM command VARY NET,INACT, to be issued to all nodes still connected to IMS, except the master terminal.

If the `/IDLE NODE` command is entered prior to IMS entering shutdown, the `/ACT` command can be entered to reactivate the node in VTAM. If the `/IDLE NODE` command is entered while in IMS shutdown, a `VARY NET,ACT` command might have to be entered directly to VTAM. However, before using the `/IDLE` command, a `/CLSDST` and then a `/CLSDST FORCE` command should be tried.

The node parameter can be generic. The generic parameter specifies nodes that already exist.

Examples

Example 1 for /IDLE Command

Entry ET:

```
/IDLE LINE 14 24
```

Response ET:

```
DFS058I  IDLE COMMAND COMPLETED
```

Explanation: The I/O on lines 14 and 24 is forced to terminate.

Example 2 for /IDLE Command

Entry ET:

```
/IDLE LINE 4 NOSHUT
```

Response ET:

```
DFS058I  IDLE COMMAND COMPLETED
```

Explanation: The I/O on line 4, which is a 3270 remote BTAM line, is forced to terminate.

Example 3 for /IDLE Command

Entry ET:

```
/IDLE LINK 2
```

Response ET:

```
DFS058I  IDLE COMMAND COMPLETED
```

Explanation: The input/output on logical link 2 is forced to terminate.

Example 4 for /IDLE Command

Entry ET:

```
/IDLE NODE ABC
```

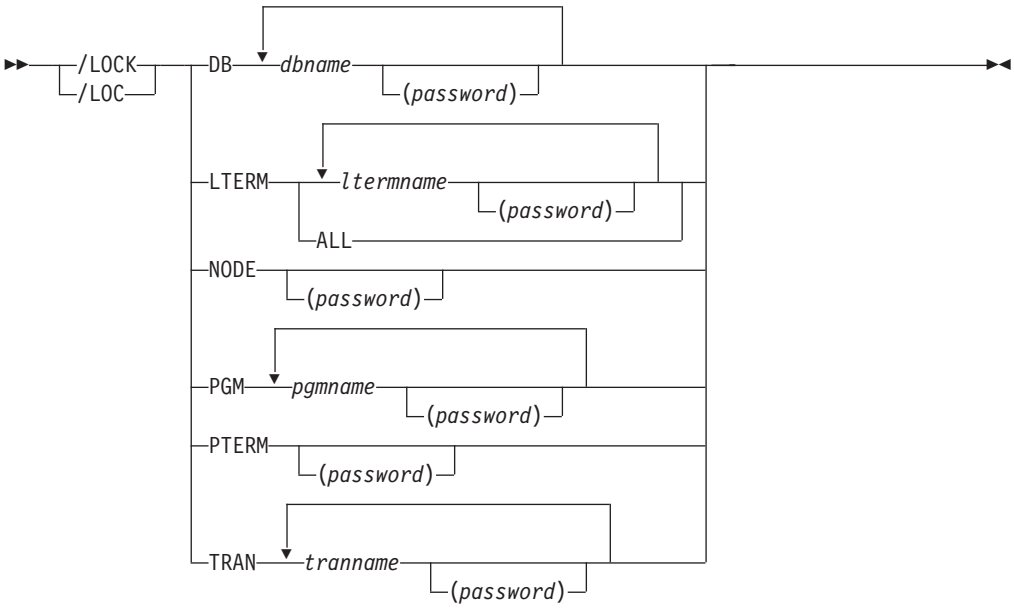
Response ET:

```
DFS058I  IDLE COMMAND COMPLETED
```

Explanation: A `VARY NET, INACT` command is executed that will cause the IMS VTAM LOSTERM exit to be entered with an indication that the session has been terminated.

Chapter 30. /LOCK

Format



Environments and Keywords

Table 66 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 66. Valid Environments for the /LOCK Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/LOCK	X	X	X
DB	X	X	
LTERM	X		X
NODE	X		X
PGM	X	X	X
PTERM	X		X
TRAN	X		X

Usage

/LOCK stops the sending and receiving of messages to and from a terminal, stops the scheduling of messages containing a specific transaction code, stops the scheduling of a specific program, and stops the use of a database. Queuing of output messages for a particular communication line, terminal, or transaction code is not affected.

The /LOCK command can be used on HALDBs. For more information see Appendix H, “High Availability Large Database Commands,” on page 867

© Copyright IBM Corp. 1974, 2003

The output of the /LOCK DB command is changed when the command is entered through the OM API. In this case, the DFS058I message is not returned to OM. The command response returned to OM contains one or more of the following messages: DFS0488I, DFS3466I, or DFS132. The OM API does not support the /LOCK command without any keywords.

You can specify password security on the /LOCK command by requiring a password after a resource name is entered. If the resource is not defined with password protection in SMU or SMU security is not used, the password is ignored.

A resource name can also be defined with password protection in SAF. If the parameter, LOCKSEC=Y (N is the default) is specified on the DFSDCxxx IMS.PROCLIB member, the SAF and user exit calls are made after the SMU calls. If the resource is not defined to SAF, or is defined and is authorized to the user, the command is processed. If the resource is defined to SAF but not authorized for use, the command is rejected with a DFS3689W message.

DATABASE
LOCK prevents subsequently scheduled programs from accessing the database. /LOCK DATABASE does not close the database or affect currently scheduled programs.

If the database is a DEDB or MSDB, programs using the database will not be scheduled. For other databases, the programs will still be scheduled. If the INIT call was issued, however, a call against the database will result in either a 3303 pseudo abend or a BA status code.

For DBCTL, CCTL can specify LONG or SHORT when it schedules a PSB. If the database is currently scheduled to a LONG thread, the command is rejected. If not, the thread completes before the database is acted upon. If the thread completes, a commit point or transaction termination is the result.

For the results of issuing this command on a shared secondary index, see Appendix D, “Shared Secondary Index Database Commands,” on page 849

LTERM
Specifies the logical terminal for which sending and receiving of messages is to be stopped. /LOCK LTERM applies only to logical terminals associated with the entering physical terminal.

The password specified with /LOCK LTERM is ignored for dynamic LTERMs because the Security Maintenance Utility is used for password support and is not supported for dynamic terminals. The /LOCK command itself is not ignored. The password is also ignored for static LTERMs when SMU security is not used or the LTERM is not defined with password security.

The /LOCK LTERM ALL command can only be used when none of the logical terminals associated with the entering physical terminal have passwords.

NODE
Specifies that the sending and receiving of messages for the VTAM node associated with the entering physical terminal is to be stopped.

The password specified with /LOCK NODE is ignored for dynamic ACF/VTAM nodes because the Security Maintenance Utility is used for password support and is not supported for dynamic terminals. The /LOCK command itself is not ignored. The password is also ignored for static nodes when SMU security is not used or the node is not defined with password security.

PROGRAM
Specifies the program for which scheduling is to be stopped.

PTERM
Specifies that the sending and receiving of messages for the entering physical terminal is to be stopped.

TRANSACTION
Specifies that scheduling of messages containing this transaction code is to be stopped. The /LOCK TRANSACTION command cannot be used with Fast Path exclusive transactions but can be used with Fast Path potential transactions.

/LOCK TRANSACTION cannot be used for CPI Communications driven programs.

/LOCK LTERM, NODE, and PTERM apply only to the entering physical terminal. /LOCK DATABASE | PROGRAM | TRANSACTION is only valid if entered from the master terminal, the system console, a TCO script, or an AOI application program.

Issuing /LOCK for a physical terminal, logical terminals, or nodes prevents application program output from being sent (assuming nonresponse mode transactions) and message switches from any other terminals. Output message queuing continues for the locked physical terminal. /STOP LTERM can be used instead of /LOCK LTERM, because both stop the sending and receiving of messages to the physical terminal.

If the terminals are on a switched communication network and a physical or logical terminal disconnection occurs, an implied /UNLOCK is processed against the physical terminal and inquiry logical terminal.

The /LOCK command cannot be used with logical terminals assigned to deallocated users.

Examples

Example 1 for /LOCK Command

Entry ET:

```
/LOCK DATABASE TREEFARM
```

Response ET:

```
DFS058I  LOCK COMMAND COMPLETED
```

Explanation: Application programs that use the MSDB database named TREEFARM are no longer scheduled. Transaction input can continue.

Example 2 for /LOCK Command

Entry ET:

```
/LOCK LTERM ALL
```

Response ET:

```
DFS058I  LOCK COMMAND COMPLETED
```

Explanation: Output is queued but not sent to the logical terminals associated with the physical terminal from which /LOCK was entered.

Example 3 for /LOCK Command

Entry ET:

```
/LOCK PROGRAM APPLETRE
```

Response ET:

```
DFS058I  LOCK COMMAND COMPLETED
```

Explanation: Program APPLETRE can no longer be scheduled.

Example 4 for /LOCK Command

Entry ET:

```
/LOCK PTERM
```

Response ET:

```
DFS058I  LOCK COMMAND COMPLETED
```

Explanation: Application program output (if nonresponse mode transaction) and message switches from other terminals are not sent to the entering physical terminal. Output is queued for later transmission when the terminal is unlocked.

Example 5 for /LOCK Command

Entry ET:

```
/LOCK TRANSACTION SEED
```

Response ET:

```
DFS058I  LOCK COMMAND COMPLETED
```

Explanation: The application program for transaction code SEED can no longer be scheduled. Transaction input can continue.

Entry ET:

```
/LOCK TRANSACTION TXCDLM06
```

Response ET:

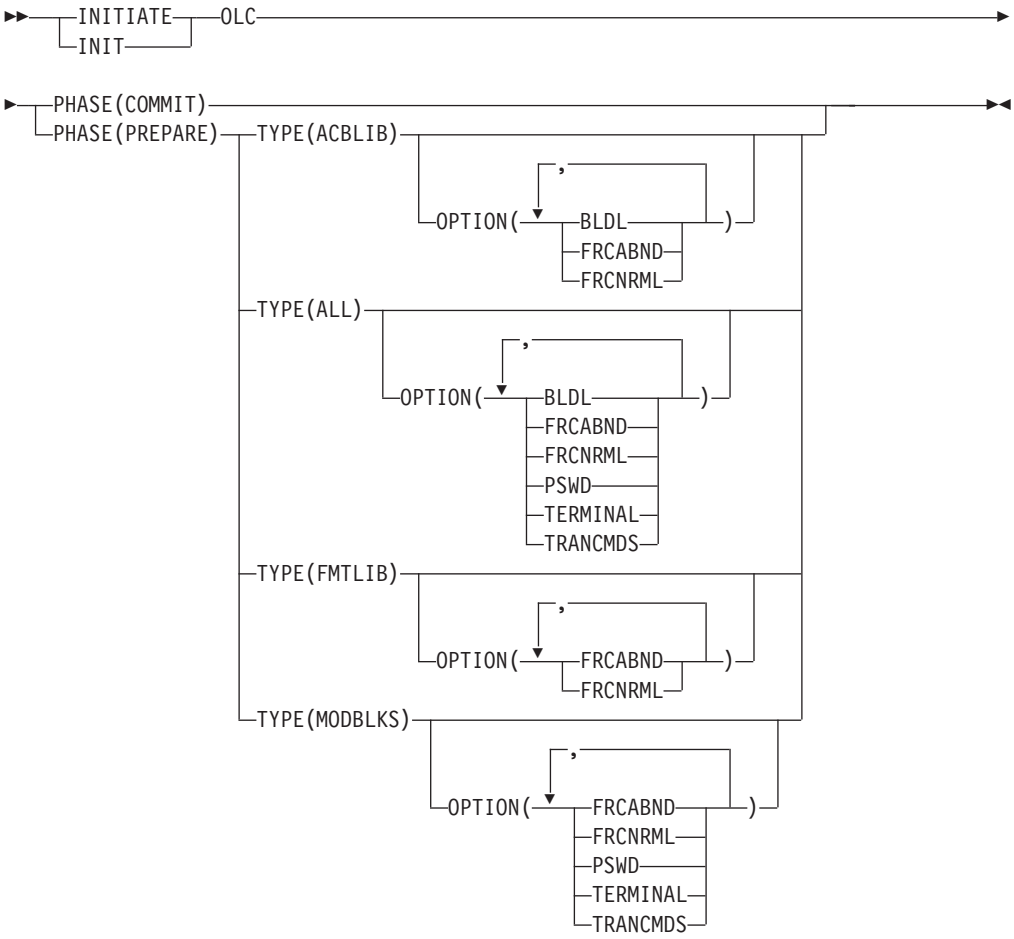
```
DFS058I  LOCK COMMAND COMPLETED EXCEPT TRANSACTION
TXCDLM06
```

Explanation: TXCDLM06 is a Fast Path exclusive transaction. /LOCK cannot stop the scheduling of Fast Path exclusive transactions.

Chapter 31. INITIATE

Format

INITIATE OLC



Environments and Keywords

Table 67 lists the environments (DB/DC, DBCTL, and DCCTL) from which the INITIATE command and keywords can be issued.

Table 67. Valid Environments for the INITIATE OLC Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
INITIATE OLC	X	X	X
ACBLIB	X	X	X
BLDL	X	X	X
FMTLIB	X		X
FRCABND	X	X	X
FRCNRML	X	X	X
MODBLKS	X	X	X

© Copyright IBM Corp. 1974, 2003

Table 67. Valid Environments for the INITIATE OLC Command and Keywords (continued)

Command / Keywords	DB/DC	DBCTL	DCCTL
OPTION	X	X	X
PHASE	X	X	X
PSWD	X	X	X
TERMINAL	X		X
TRANCMDS	X		X
TYPE	X	X	X

INITIATE OLC

If an IMSplex is running with an RM environment (RMENV=Y), the INITIATE OLC (initiate online change) command initiates a phase of global online change of resources and coordinates the phase with all of the IMSs in the IMSplex. The RM environment (RMENV = Y | N) is specified in the DFSCGxxx PROCLIB member. INITIATE OLC is similar to /MODIFY PREPARE and /MODIFY COMMIT, except that it applies to an IMSplex-wide global online change. OM sends the INITIATE OLC command to one master IMS in the IMSplex. The command master IMS uses RM services to coordinate the online change with all participating IMSs.

If the command master IMS in an IMSplex is running without an RM environment (RMENV=N), the INITIATE OLC command is performed locally at the IMS to which the command was sent. The OLCSTAT DS must be unique for each IMS in the IMSplex that is running with no RM environment (RMENV=N). The OLCSTAT DS cannot be shared between IMSs and must be unique. If the OLCSTAT DS contains the name of an IMS other than the one processing the online change, INITIATE OLC is rejected with a reason code indicating that the OLCSTAT data set is invalid for the environment. You can use the QUERY OLC command to display the contents of the OLCSTAT data set to determine which IMS member name is invalid. To correct the data set, you can use the OLC utility, DFSUOLC0.

If INITIATE OLC is issued with a route list containing either the default Route All or multiple IMSs that do not have RM running, the INITIATE OLC command must be issued separately to each IMS. To perform online change for IMSs that do not have RM, you must issue the INITIATE OLC command to each IMS. For example, if you have four IMS systems, you must issue the command to each IMS. To determine which IMSs are defined with RMENVNO in the DFSCGxxx PROCLIB member, you can issue the QUERY MEMBER SHOW(ATTRIB) command.

If multiple IMS sytems are in an IMSplex where some IMSs use RM services and other IMSs do not, you must ensure that the OLCSTAT data sets are properly defined to the IMS systems. The IMS systems that use RM services and global online change must share one OLCSTAT DS. Each IMS system that does not use RM services must have its own unique OLCSTAT DS.

INITIATE OLC is not supported if local online change is enabled. The INITIATE OLC PHASE(PREPARE) command is rejected if the IMS to which the command is routed does not support global online change. If this occurs and there is an IMS that supports global online change, the user must route the command to a specific IMS that supports global online change.

The correct online change command sequence is INITIATE OLC PHASE(PREPARE) followed by INITIATE OLC PHASE(COMMIT). Use the /DISPLAY MODIFY command to

display the work in progress for resources to be changed or deleted, before attempting the COMMIT phase. When COMMIT is successful, the modifications persist across all IMS restarts, unless global online change occurs while this IMS is down.

If the INITIATE OLC PHASE(PREPARE) is specified without a FRCABND or FRCNRML keyword and the command fails as one or more IMSs are down or go down before the online change is committed, the online change must be aborted and started over. Issue the TERMINATE OLC command to abort the online change.

If the INITIATE OLC PHASE(PREPARE) is specified with the FRCNRML keyword and the command fails for any IMS, you can proceed with an INITIATE OLC PHASE(COMMIT) command, if desired, after shutting down those IMSs where prepare failed. Otherwise, you must abort the online change and start over.

If the INITIATE OLC PHASE(PREPARE) is specified with the FRCABND keyword and the command fails for any IMS, you can proceed with an INITIATE OLC PHASE(COMMIT) command, if desired, after cancelling those IMSs where prepare failed. Otherwise, you must abort the online change and start over.

The INITIATE OLC command master usually performs the online change phase locally first. If the online change phase fails locally, the command master usually skips sending the online change phase to the other IMSs, sets a completion code for each other IMS indicating that the online change phase was not attempted, and terminates command processing. However, if the INITIATE OLC PHASE(COMMIT) command fails on the local IMS because of work in progress, the command master still sends the commit phase 1 to the other IMSs. The purpose is to report work in progress for all the IMSs in the IMSplex, to facilitate completion of work in progress.

In a mixed IMSplex, you might have some variations of IMSs that support a particular type of online change and some that do not. At least one IMS in the IMSplex must successfully perform the online change phase for the command to be considered successful. If no IMS in the IMSplex supports the type of online change that the command is attempting, the command reason code indicates that none of the IMSs performed the online change phase. If you enter an INITIATE OLC PHASE(PREPARE) command that does not apply to any IMS in the IMSplex, you must terminate the online change with a TERMINATE OLC command.

If the INITIATE OLC PHASE(COMMIT) command fails for any IMS before the OLCSTAT data set is updated, you may either correct the errors and try the commit again or abort the online change with a TERMINATE OLC command.

If the INITIATE OLC PHASE(COMMIT) command fails for any IMS after the OLCSTAT data set has been updated, you may correct the errors and try the commit again. The online change cannot be aborted.

If an IMS abends during online change and the INITIATE OLC PHASE(PREPARE) command was not specified with FRCABND, then issue the TERMINATE OLC command to abort the online change. The INITIATE OLC PHASE(COMMIT) command is not permitted in this case. If an IMS abends during online change and the INITIATE OLC PHASE(PREPARE) command was specified with FRCABND, then the INITIATE OLC PHASE(COMMIT) command is permitted.

IMSplex commands and classic commands that come from the OM interface are rejected during the commit phase, if the command changes resources. Commands that change resources could interfere with the online change of the resources.

IMSplex commands and classic commands that come from the OM interface are permitted during the commit phase, if the command displays resources, such as QUERY or /DISPLAY. Classic commands that are entered from the system console or an IMS terminal are queued during the online change commit phase. These commands run after the online change is committed or aborted.

The resources that may be changed online are areas, databases, DMBs, Fast Path DEDB databases, MFS formats, programs, routing codes, SMU security definitions, and transactions.

If IMS is in an IMSplex and Resource Manager (RM) is using a resource structure, the transactions being added by the online change are registered to RM if they are not already registered to RM. If any error occurs during the CSLRMUPD request to register the transactions, the /MODIFY or INITIATE OLC command will fail and IMS will be in a COMMIT2F state.

If an INITIATE OLC PHASE(COMMIT) command results in the IMS remaining in a COMMIT2F state, a completion code of the error that caused it is returned to OM. If the error can be fixed, the INITIATE OLC PHASE(COMMIT) can be re-issued and CSLRMUPD retried and the online change completed. If the error cannot be fixed, the IMS has to be cancelled and warm started as the online change is not complete

This command is invalid on an XRF alternate, RSR tracker, and FDR system.

Each IMS participating in the global online change does not issue the same synchronous online change messages to the master terminal or system console that it does for a local online change. The OM command response contains information equivalent to the online change messages that appear for the local online change, such as the DFS3499 message contents. See “INITIATE Return and Reason Codes” on page 394 for more information about what is returned from an INITIATE OLC command.

Each IMS participating in the global online change may issue asynchronous online change messages to the system console, such as DFS3400, DFS3445, and DFS3498.

The OM command time-out default of 300 seconds (5 minutes) may not be enough time for the online change phase to complete. It may be required to specify a time-out value on the command based on the needs of the installation.

OPTION()

Specifies an additional function to be performed during an online change. Following is a list of additional functions:

BLDL Sends asynchronous information message DFS3498I to the system console for each PSB or DMB not found in ACBLIB. This message does not affect the success of the command.

FRCABND

Forces the online change prepare phase even if one or more required IMSs in the IMSplex have abended. An IMS is required to participate in the global online change if it is listed in the OLCSTAT data set as being current with the online change libraries. An IMS is listed in the OLCSTAT data set because it either participated in the last global online change or coldstarted since the last global online

change. A successful global online change specified with the FRCABND removes an abended IMS from the list of required IMSs in the OLCSTAT data set.

FRCNRML
Forces the online change prepare phase even if one or more required IMSs in the IMSplex have terminated normally. An IMS is required to participate in the global online change if it is listed in the OLCSTAT data set as being current with the online change libraries. An IMS is listed in the OLCSTAT data set because it either participated in the last global online change or coldstarted since the last global online change. A successful global online change specified with the FRCNRML keyword removes an IMS that shut down normally from the list of required IMSs in the OLCSTAT data set.

PSWD Specifies specific types of SMU password security definitions to add, if they had not existed before. The application of new security definitions are always attempted based upon the prevailing security options in effect from the last IMS initialization.

TERMINAL
Specifies specific types of SMU terminal security definitions to add, if they had not existed before. The application of new security definitions are always attempted based upon the prevailing security options in effect from the last IMS initialization.

TRANCMDS
Specifies specific types of SMU transaction command security definitions to add, if they had not existed before. The application of new security definitions are always attempted based upon the prevailing security options in effect from the last IMS initialization.

PHASE()
Refers to the online change phase to start. The PREPARE phase must be performed first, followed by the COMMIT phase:

PREPARE
Performs the online change prepare phase on each IMS system listed in the OLCSTAT data set by specifying which resources are to be added, changed, and deleted. PREPARE is not recovered across an emergency restart and must be reentered after restart if the COMMIT phase did not complete prior to IMS failure.

The prepare phase is rejected if any IMSs in the IMSplex current with the online change libraries are down, unless the FRCABND or the FRCNRML option is used. The OLCSTAT data set lists the IMSs that are current with the online change libraries. These IMSs either participated in the last global online change or coldstarted after the last global online change. To force a global online change in spite of abended IMSs, use the FRCABND option. To force a global online change in spite of IMSs that are shut down normally, use the FRCNRML option. The prepare phase fails if any IMS participating in the global online change prepare phase fails, detects an error, or fails to respond in time. Zero, one, or more of the IMSs remain in an online change prepare complete state. Issue the `QUERY MEMBER TYPE(IMS) SHOW(STATUS)` command to display the online change state of all the IMSs in the IMSplex. Evaluate the `QUERY MEMBER TYPE(IMS)` output to help you decide whether to issue

the INITIATE OLC PHASE(PREPARE) command to try the prepare again, or to issue the TERMINATE OLC command to abort the online change. The IMSs that are in an online change state remain in an online change state until you abort or commit the online change. IMS does not automatically abort online change because of a failure. IMS leaves the IMSs in their online change states. You must issue the QUERY MEMBER TYPE(IMS) SHOW(STATUS) command and evaluate the output to decide whether to abort the online change or attempt the online change phase again.

COMMIT

Performs the online change commit phase on each IMS listed in the OLCSTAT data set, which commits the online changes by bringing all the newly defined resources online, updating changed resources, and removing deleted resources. The commit phase consists of commit phase 1, the OLCSTAT data set update, commit phase 2, and commit phase 3. The OLCSTAT data set is updated with the new current online change libraries and the list of IMSs that are current with the current online change libraries. The commit phase 2 switches the online environment from the active ACBLIB, FORMAT, MATRIX, or MODBLKS libraries to the inactive libraries containing the new or changed resource descriptions.

The commit phase fails if any IMS participating in the global online change commit phase fails, detects an error, or fails to respond in time. Zero, one, or more of the IMSs might be in an online change prepare complete state, a commit phase 1 complete state, or no longer in an online change state because commit phase 2 succeeded. Issue the QUERY MEMBER TYPE(IMS) SHOW(STATUS) command to display the online change state of all the IMSs in the IMSplex. Evaluate the QUERY MEMBER TYPE(IMS) output to help you decide whether to issue the INITIATE OLC PHASE(COMMIT) command to try the commit again, or issue the TERMINATE OLC command to abort the online change. The IMSs that are in an online change state remain in an online change state until you abort or commit the online change.

IMS does not automatically abort online change because of a failure. IMS leaves the IMSs in their online change states. You must issue the QUERY MEMBER TYPE(IMS) SHOW(STATUS) command and evaluate the output to decide whether to abort the online change or attempt the online change phase again.

If a database is deleted, IMS closes the database and makes it unavailable to programs. Also, if the database is authorized for DBRC, COMMIT unauthorizes it.

If FMTLIB is being changed, most new and in-progress requests are queued for processing after the INITIATE OLC PHASE(COMMIT) command is completed.

Most new and in-progress requests are queued for processing.

An INITIATE OLC PHASE(COMMIT) is rejected if:

- An INITIATE OLC PHASE(PREPARE) command was not previously entered.
- Commit phase 1 fails on one or more IMSs in the IMSplex.
- Transactions to be deleted have messages queued.

- Transactions to be changed by the following system definition keywords have messages queued:
  - MSGTYPE
  - INQUIRY
  - FPATH
  - EDIT
  - SPA
- Transactions to be changed by keywords not shown in the previous bullet have messages queues and the transactions have not been the object of a /STOP or /PSTOP command.
- Transactions with access to programs or databases to be changed or deleted are prevented from updating and the transactions have not been the object of a /STOP or /PSTOP command. When a transaction shows a status of USTOP during a /DISPLAY TRAN command, the /STO command might need to be entered because the /PSTOP command might not allow the INITIATE OLC command to complete the transaction.
- Programs or databases to be changed or deleted are scheduled.
- Any program (PSB) that is currently scheduled has a program, database, or transaction change or deletion. This includes Wait for Input (WFI) and Fast Path transactions. All WFI and Fast Path regions that reference changed or deleted routing codes, programs, or databases must be stopped before entering the INITIATE OLC PHASE(COMMIT).
- An I/O error occurs while reading the inactive MFS FORMAT library directory or the resident index ($$IMSDIR) member.
- A /START DATABASE command is in progress for any database that is changed or deleted.
- A /DBDUMP or /DBRECOVERY command is in progress for any database that is changed or deleted.
- An IMS that participated in the prepare phase shut down normally or abended. The online change must be aborted and started over.
- OM routed the command to an IMS where local online change is enabled.
- The OLCSTAT data set is not locked for global online change.

  The INITIATE OLC PHASE(PREPARE) commands locks the OLCSTAT data set, so that no IMSs can initialize during the global online change. The commit command is rejected if the lock is not set. The INITIATE OLC PHASE(PREPARE) command might not have completed successfully or the OLCSTAT data set contents might be invalid.
- An IFP region is running. INITIATE OLC PHASE(COMMIT) is rejected because an active route code exists. All IFP regions must be terminated before issuing a COMMIT.

Operator action might be required to prevent the preceding conditions. For example, if a program or database is scheduled, the operator should either wait until the program or database is finished before the INITIATE OLC PHASE(COMMIT) command is entered or the operator should issue /STOP, /PSTOP, or UPDATE TRAN for the associated transaction.

TYPE()
Specifies the types of resources that are changed online. The resources are as follows:

ACBLIB
Causes IMS to prepare to add, change, or delete database descriptors (DMB) and program descriptors (PSB) to ACBLIB. The online environment is switched from the active ACBLIB to the inactive ACBLIB containing new or changed DMB and PSBs. ACBLIB members that are not found are displayed as CHANGED because a DDIR or PDIR exists in MODBLKS. Additions or deletions to ACBLIB require the MODBLKS parameter. Quiesces transactions to be changed or deleted and transactions that access databases or programs to be changed or deleted. Fast Path DEDBs are supported but Fast Path MSDBs are not supported.

ALL Prepares to add, change, and delete resources for the ACBLIB, FMTLIB, or MODBLKS libraries. The BLDL function is performed for ACBLIB.

FMTLIB
Causes IMS to prepare to add, change, or delete MFS maps in the FMTLIB library that are produced by the MFS Language utility and Service utility. The online environment is switched from the active FMTLIB to the inactive FMTLIB containing new MFS maps produced by the MFS Language utility and Service utility. The FMTLIB parameter is ignored with DBCTL. There are no restrictions on the use of MFS format blocks until the `INITIATE OLC PHASE(COMMIT)` command is entered.

MODBLKS
Causes IMS to prepare to add, change, or delete database, transaction, program, or RTCODE resources in the MODBLKS library or SMU security definitions in the MATRIX library. The online environment is switched from the active MODBLKS and MATRIX libraries to the inactive MODBLKS and MATRIX libraries. The MODBLKS library contains a subset of control blocks produced by IMS system definition and SMU security definitions. The MATRIX library contains SMU security definitions for resources defined in the MODBLKS library. Transactions to be changed or deleted and transactions that access databases or programs to be changed or deleted are quiesced.

INITIATE OLC Output Fields

Table 68 on page 393 shows the output fields for an `INITIATE OLC` command. The columns in the table are as follows:

Short Label
Contains the short label generated in the XML output.

Keyword
Identifies the keyword on the command that caused the field to be generated. N/A appears for output fields that are always returned.

Meaning
Provides a brief description of the output field.

Table 68. INITIATE OLC Output Field Descriptions

SHORT LABEL	KEYWORD	MEANING
MBR	N/A	IMSplex member that built the line of output. IMS identifier of the IMS that is the master of this online change phase. IMS identifier is always returned.
IMSMBR	N/A	IMS member that performed the global online change phase. The IMS member name is always returned.
CC	N/A	Completion code returned by IMS member that performed the global online change phase. Completion code is always returned.
ERRT	N/A	Error text returned by IMS member that failed performing the global online change phase. Error text may be returned if the completion code is nonzero.
ACBL	N/A	Current ACBLIB library suffix (if prepare succeeded or commit succeeded that did not include ACBLIB), or new ACBLIB library suffix (if commit succeeded for ACBLIB). The ACBLIB suffix is returned if the online change prepare phase or commit phase is successful. Suffix A means that ACBLIBA is current. Suffix B means that ACBLIBB is current.
FMTL	N/A	Current FMTLIB library suffix (if prepare succeeded or commit succeeded that did not include FMTLIB), or new FMTLIB library suffix (if commit succeeded for FMTLIB). The FMTLIB suffix is returned if the online change prepare phase or commit phase is successful. Suffix A means that FMTLIBA is current. Suffix B means that FMTLIBB is current.
MODB	N/A	Current MODBLKS library suffix (if prepare succeeded or commit succeeded that did not include MODBLKS), or new MODBLKS library (if commit succeeded for MODBLKS). The MODBLKS suffix is returned if the online change prepare phase or commit phase is successful. Suffix A means that MODBLKSA and MATRIXA are current. Suffix B means that MODBLKSB and MATRIXB are current.
MODI	N/A	Current modify id (if prepare succeeded), or new modify id (if commit succeeded). The modify id - 1 indicates the number of global online changes that have been performed. The modify id is returned if the online change prepare phase or commit phase is successful.

INITIATE Return and Reason Codes

Two sets of return and reason codes are returned when a command is issued through the OM API. One set of codes is for the OM request and the other set of codes is for the command itself. The OM request return and reason codes that can be returned as a result of the INITIATE OLC command are standard for all commands entered through the OM API.

The return and reason codes returned by the INITIATE OLC command are passed from the OM API on to the SPOC application. These codes are encapsulated in XML tags similar to how the response to the command is encapsulated in XML tags. The IBM-supplied TSO SPOC displays these codes, whereas a user-written SPOC application might choose not to display these codes.

Some reason codes are accompanied by a complete list of IMSs and return codes. The reason code meaning indicates whether a list is returned. A partial list of IMSs and return codes might be returned with any INITIATE OLC error reason code, if any output was built before the error was detected.

Table 69 shows the return and reason codes that might be returned by the INITIATE OLC command as well as the meaning of the codes.

Table 69. Return and Reason Codes for INITIATE OLC

Return Code	Reason Code	Meaning
X'00000000'	X'00000000'	The INITIATE OLC command completed successfully. The INITIATE OLC command applies to all of the IMSs listed in the OLCSTAT data set. Each IMS listed in the OLCSTAT data set is current with the online change libraries and required to participate in the online change. Neither FRCABND nor FRCNRML was specified. If the INITIATE OLC PHASE(PREPARE) command was specified, all of the IMSs listed in the OLCSTAT data set are now in an online change prepare state. If the INITIATE OLC PHASE(COMMIT) command was specified, all of the IMSs listed in the OLCSTAT data set successfully committed the online change. An output line is built for each IMS listed in the OLCSTAT data set. Each output line contains the IMS member name and a completion code of zero.
X'00000004'	X'0000100C'	The command completed successfully, but was not applicable to one or more IMSs for acceptable reasons. The INITIATE OLC command applies to all of the IMSs listed in the OLCSTAT data set. Each IMS listed in the OLCSTAT data set is current with the online change libraries and required to participate in the online change (unless FRCABND or FRCNRML is specified). An output line is built for each IMS listed in the OLCSTAT data set. Each output line contains the IMS member name and a completion code. A nonzero completion code may be accompanied by error text. One or more of the IMSs contain a completion code indicating the online change phase did not apply to this IMS, such as the IMS state is shutdown and FRCSHUT was specified, the IMS state is abended and FRCABND was specified, the online change type does not apply to this IMS, or this IMS is already in the correct online change state. The INITIATE OLC completion code table contains the list of completion codes and error text that can be returned by the INITIATE OLC command.

Table 69. Return and Reason Codes for INITIATE OLC (continued)

Return Code	Reason Code	Meaning
X'00000008'	X'00002040'	The INITIATE OLC command is rejected because of an invalid filter. The INITIATE OLC PHASE(PREPARE) OPTION(BLDL) can only be specified with TYPE ACBLIB or ALL. The PSWD, TERM, and TRANCMDS options can only be specified with TYPE MODBLKS or ALL. The INITIATE OLC PHASE(COMMIT) command does not support TYPE or OPTION.
X'0000000C'	X'00003000'	The INITIATE OLC command is successful for at least one IMS but not all IMSs. The INITIATE OLC command applies to all of the IMSs listed in the OLCSTAT data set. Each IMS listed in the OLCSTAT data set is current with the online change libraries and required to participate in the online change (unless FRCABND or FRCNRML is specified). An output line is built for each IMS listed in the OLCSTAT data set. Each output line contains the IMS member name and a completion code. A nonzero completion code may be accompanied by error text. One or more of the IMSs returned an error completion code. The INITIATE OLC completion code table contains the list of completion codes and error text that can be returned by the INITIATE OLC command. If INITIATE OLC PHASE(PREPARE) fails, the online change must be aborted and started over. If INITIATE OLC PHASE(COMMIT) fails and you want to retry the commit, correct the problem and issue the INITIATE OLC PHASE(COMMIT) command again. To abort the online change, issue the TERMINATE OLC command. If the INITIATE OLC PHASE(PREPARE) command failed or the INITIATE OLC PHASE(COMMIT) command failed before the commit master updated the OLCSTAT data set, the online change is aborted. See 'INITIATE OLC error handling' for more details.
X'0000000C'	X'00003004'	The INITIATE OLC command failed for all of the IMSs. The INITIATE OLC command applies to all of the IMSs listed in the OLCSTAT data set. Each IMS listed in the OLCSTAT data set is current with the online change libraries and required to participate in the online change (unless FRCABND or FRCNRML is specified). An output line is built for each IMS listed in the OLCSTAT data set. Each output line contains the IMS member name and a completion code. A nonzero completion code may be accompanied by error text. The INITIATE OLC completion code table contains the list of completion codes and error text that can be returned by the INITIATE OLC command. If INITIATE OLC PHASE(PREPARE) fails for all of the IMSs listed in the OLCSTAT data set, no IMS is in an online change state. You may try the online change again with another INITIATE OLC PHASE(PREPARE) command. If INITIATE OLC PHASE(COMMIT) fails and you want to retry the commit, correct the problem and issue the INITIATE OLC PHASE(COMMIT) command again. To abort the online change, issue the TERMINATE OLC command. If the INITIATE OLC PHASE(PREPARE) command failed or the INITIATE OLC PHASE(COMMIT) command failed before the commit master updated the OLCSTAT data set, the online change is aborted. See 'INITIATE OLC error handling' for more details.

Table 69. Return and Reason Codes for INITIATE OLC (continued)

Return Code	Reason Code	Meaning
X'00000010'	X'00004004'	The INITIATE OLC command failed because there is no CQS. RM attempted to access the process resource on the resource structure, but it failed because CQS is not available. The online change phase may have succeeded on one or more IMSs. See 'INITIATE OLC error handling' for more details.
X'00000010'	X'0000400C'	The INITIATE OLC command failed because it is invalid for an XRF alternate.
X'00000010'	X'00004014'	The INITIATE OLC command failed because it is invalid for an RSR tracker.
X'00000010'	X'0000401C'	The INITIATE OLC command failed because it is invalid for an FDR region.
X'00000010'	X'00004018'	The INITIATE OLC command failed because the RM resource structure is not available. The online change phase may have succeeded on one or more IMSs. See 'INITIATE OLC error handling' for more details.
X'00000010'	X'00004100'	The INITIATE OLC PHASE(PREPARE) command is rejected because the resource structure is full. RM failed trying to create the process resource on the resource structure.
X'00000010'	X'00004104'	The INITIATE OLC command failed because RM is not available. The online change phase may have succeeded on one or more IMSs. Either there is no RM address space, or RM is active but not registered to SCI because CQS or the resource structure is not available. See 'INITIATE OLC error handling' for more details.
X'00000010'	X'00004108'	The INITIATE OLC command failed because SCI is not available. The online change phase may have succeeded on one or more IMSs. See 'INITIATE OLC error handling' for more details.
X'00000010'	X'0000410C'	The INITIATE OLC command is rejected, because global online change is not enabled. Local online change is enabled. Use the /MODIFY command for local online change. If your IMSplex is made up of some IMSs that support global online change and some that support local online change, route the INITIATE OLC command to an IMS that is enabled for global online change.
X'00000010'	X'00004110'	The INITIATE OLC command is rejected, because the command does not apply to the online change state of the command master. INITIATE OLC PHASE(COMMIT) is rejected if the command master is not in an online change prepare state. INITIATE OLC PHASE(COMMIT) is rejected if routed to an IMS that already successfully completed commit phase 3. INITIATE OLC PHASE(PREPARE) is rejected if the command master is already in an online change prepare state. See 'INITIATE OLC error handling' for more details.
X'00000010'	X'00004111'	The INITIATE OLC command failed because the command master is not in the OLCSTAT data set.
X'00000010'	X'00004114'	The INITIATE OLC command failed because of an error accessing the OLCSTAT data set. The online change phase may have succeeded on one or more IMSs. A DFS2843 message is sent to the OM output exit as unsolicited output. See 'INITIATE OLC error handling' for more details.

Table 69. Return and Reason Codes for INITIATE OLC (continued)

Return Code	Reason Code	Meaning
X'00000010'	X'00004118'	The INITIATE OLC command failed because of an error allocating the OLCSTAT data set. The online change phase may have succeeded on one or more IMSs. A DFS2848 message is sent to the OM output exit as unsolicited output. See 'INITIATE OLC error handling' for more details.
X'00000010'	X'0000411C'	The INITIATE OLC command failed because of an error in the OLCSTAT data set contents. One or more of the values is invalid. A DFS2844 message is sent to the OM output exit as unsolicited output.
X'00000010'	X'00004120'	The INITIATE OLC command is rejected because an online change command is already in progress on this IMS, which may be INITIATE OLC, TERMINATE OLC, or /DISPLAY MODIFY.
X'00000010'	X'00004124'	The INITIATE OLC PHASE(PREPARE) command is rejected because another process of the same type is already in progress. Wait until the other process has completed. If QUERY MEMBER TYPE(IMS) shows no IMS in an online change state, the process resource is residual after an online change error. Clean up the process resource with a TERMINATE OLC command.
X'00000010'	X'0000412C'	The OLCSTAT data set contains the name of an IMS other than the IMS processing the online change. Use DFSUOLC0 to correct the data set.
X'00000014'	X'00005000'	The INITIATE OLC command is rejected because an IMODULE GETSTOR storage request failed.
X'00000014'	X'00005004'	The INITIATE OLC command failed because a DFSOCMD response buffer could not be obtained. The online change phase may have succeeded on one or more IMSs. See 'INITIATE OLC error handling' for more details.
X'00000014'	X'0000500C'	The INITIATE OLC command failed because an AWE could not be obtained.
X'00000014'	X'00005100'	The INITIATE OLC command failed because of an RM error. The online change phase may have succeeded on one or more IMSs. The RM error may be due to a resource structure failure that causes RM to lose knowledge of an IMSplex-wide process. See 'INITIATE OLC error handling' for more details.
X'00000014'	X'00005104'	The INITIATE OLC command failed because of a CQS error. The online change phase may have succeeded on one or more IMSs. See 'INITIATE OLC error handling' for more details.
X'00000014'	X'00005108'	The INITIATE OLC command failed because of an SCI error. The online change phase may have succeeded on one or more IMSs. See 'INITIATE OLC error handling' for more details.
X'00000014'	X'00005FFF'	The INITIATE OLC command failed because of an internal IMS error. The online change phase may have succeeded on one or more IMSs. See 'INITIATE OLC error handling' for more details.

INITIATE Completion Codes

The INITIATE command (with the OLC keyword) can result in errors that leave one or more of the IMSs in the IMSplex in various online change states. Table 70 on page 398

page 398 contains the possible completion codes that can be returned as a result of issuing an INITIATE OLC command. The table also includes a brief explanation of the code and provides the error text if applicable.

Table 70. Completion Codes for the INITIATE OLC Command

Completion Code	Meaning	ERROR TEXT (upper case)
0	The online change prepare phase or commit phase completed successfully.	
1	The online change type does not apply to this IMS. For example, an FMTLIB online change does not apply to a DBCTL IMS. This IMS keeps status indicating that it is participating in the online change. This IMS is included in the OLCSTAT data set IMS list after a successful commit phase 1.	
2	The online change phase was not attempted by this IMS for one of the following reasons: • The online change phase master encountered an error and did not direct this IMS to perform the online change phase. • The online change phase master rejected the online change because one or more IMSs are down and the FRCNRML or FRCABND keyword was not specified on the INITIATE OLC PHASE(PREPARE) command. The online change phase master did not direct this IMS to perform the online change phase.	
3	This IMS is already in the correct online change state. The INITIATE OLC command must have been retried after a previous INITIATE OLC command resulted in a mix of successes and failures. This IMS keeps status indicating that it is participating in the online change. This IMS is included in the OLCSTAT data set IMS list after a successful commit phase 1.	
4	The online change commit phase is incomplete on this IMS. One or more online change commit phases have been completed on this IMS. All online change commit phases were not sent to this IMS due to errors.	
57	The queue structure is unavailable. IMS is unable to check if there is a global queue count for transactions to be added, changed, or delted. Commit is not permitted to proceed. The structure may be unavailable for one of the following reasons: • CQS is unavailable • The MSGQ structure failed and it hasn't been rebuilt yet • CQS lost connectivity to the MSGQ structure	THE QUEUE STRUCTURE IS UNAVAILABLE.

Table 70. Completion Codes for the INITIATE OLC Command (continued)

Completion Code	Meaning	ERROR TEXT (upper case)
58	An IMS is not registered to RM. An OLCSTAT dataset contains an IMS that is not registered to RM. Initiate fails for that IMS. If the `INITIATE OLC PHASE(PREPARE)` command is issued to an IMS that is registered to RM and the OLCSTAT data set contains an IMS system that is not registered to RM, the command fails. The unregistered IMS command line response is cc=58. To initiate online change, you must remove the unregistered IMS from the OLCSTAT data set. To remove the IMS, you need to perform the following steps: 1. Issue `QRY MEMBER TYPE(IMS) SHOW(ATRIB)` or `SHOW(ALL)` to verify the IMS that received the cc=58 was initialized with RMENV=N. 2. Unlock the OLCSTAT data set using the DFSUOLC0 utility. When online change is in progress it is necessary for the utility to delete the invalid member name. 3. Delete the invalid IMSID from the OLCSTAT data set using the DFSUOLC0 utility. 4. Issue `TERM OLC` to terminate the OLC with the valid IMSs. 5. Re-issue the OLC.	
60	IMODULE GETMAIN storage error.	
61	BCB storage error.	
62	HIOP storage error.	
63	WKAP storage error.	
70	Module load error.	Module name (8 char), module type (8 char). The module type can be 'SECURITY'.
71	Module locate error.	Module name (8 char), module type (8 char). The module type can be 'SECURITY'.
72	Randomizer load error.	FP area randomizer name (8 char).

Table 70. Completion Codes for the INITIATE OLC Command (continued)

Completion Code	Meaning	ERROR TEXT (upper case)
80	Data set error.	Function (8 char), ddname (8 char), return code (8 bytes), and error detail (8 char). Function can be one of the following: • BLDL Data set BLDL error. • BUSY The data set is busy. The detail contains the jobname that has the data set. • DIR Data set directory error. • EMPTY Library is empty. • ENQUEUE Data set enqueue error. • EOF Data set end-of-file (EOF) error. • OPEN Data set open error. Error detail can be 'DLS REG'. • READ Data set read error. • WRITE Data set write error. DDname can be ACBLIBA, ACBLIBB, FMTLIBA, FMTLIBB, MATRIXA, MATRIXB, MODBLKSA, MODBLKSB, or MODSTAT. Return code is the data set service return code. Reason code is the data set service reason code.
90	Internal error	Module name that detected internal error (8 char), unused (8 char), return code or function code (8 bytes), and error detail (8 char).
91	The online change prepare phase, commit phase, or abort phase timed out before this IMS responded to the online change prepare or commit phase. The online change prepare phase or commit phase may have succeeded on this IMS. Issue `QUERY MEMBER TYPE(IMS)` to determine the online change state of this IMS.	

Table 70. Completion Codes for the INITIATE OLC Command (continued)

Completion Code	Meaning	ERROR TEXT (upper case)
B0	Resource definition error.	Resource type (8 char), resource name (8 char), error detail (16 char). The resource type can be AREA, DATABASE, DMB, PROGRAM, SECURITY, or TRAN. The resource name can be a resource name, ACBLIB or MODBLKS. The error detail can be one of the following: • CISIZE GT BSIZE An attempt to add the areaname to the DEDB failed. The area's CI size exceeds the Fast Path buffer size (BSIZ=) of the IMS online control region. • CPCTRAN CONFLICT The online change command attempted to add a transaction name that conflicts with a CPIC transaction name. Tranname specifies the transaction that already exists as a CPIC transaction. • DESC CONFLICT The online change command attempted to add a transaction name that conflicts with a descriptor name. Tranname specifies the transaction that already exists as a descriptor name. • DUPLICATE A duplicate area name was found in the named DEDB DBD defined in the IMS system. The area name, that was defined in the DD1 operand on one of the AREA statements in the DBDGEN, is defined in another DEDB DBDGEN or is defined within the DEDB DBDGEN. • INCOMPATIBLE The DMB in the inactive ACBLIB is not compatible with IMS. The level of the inactive library is not compatible with the current IMS release. • INCONSISTENT IMS detected an inconsistency between the indicated security module and the definition of the new blocks in the inactive MODBLKS data set. Either the security option was requested on the online change command or the prevailing security option is YES or FORCE. In the module name, nnnns is the specific table name, where s is the suffix. • LTERM CONFLICT The online change command attempted to add a transaction name that conflicts with a dynamic lterm name.

Table 70. Completion Codes for the INITIATE OLC Command (continued)

Completion Code	Meaning	ERROR TEXT (upper case)
B0	Resource definition error (continued).	Tranname specifies the transaction that already exists as a dynamically created LTERM. The error detail can be one of the following (continued): • MSNAME CONFLICT The online change command attempted to add a transaction name that conflicts with an MSNAME. Tranname specifies the transaction that already exists as an MSNAME. • NOT DEFINED Control block was not defined in the active library during the online prepare phase. This is an internal error that should not occur. • NO BLOCKS Control blocks were missing from the inactive library during the online prepare phase. • NO FP INSTALLED An attempt was made to add a data entry database (DEDB) to an IMS system that does not have Fast Path installed. • NO OTHREADS An attempt was made to add a data entry database (DEDB) to an IMS system that was not initialized with DEDBs. No output threads (OTHREADS) were initialized and no I/O will be possible to the added DEDB. • PARTITION EXISTS A database partition by that name already exists. • RSRC CONFLICT The online change command is attempting to add a transaction name that conflicts with another message destination resource in the Resource Manager (RM). The resource can be a LTERM, CPIC transaction, MSNAME, or descriptor. Tranname specifies the transaction that already exists as another resource type. • UNSUPPORTED MSDB The online change attempted to add an MSDB or change a DEDB to an MSDB, which is not permitted.

Table 70. Completion Codes for the INITIATE OLC Command (continued)

Completion Code	Meaning	ERROR TEXT (upper case)
B1	Resource state error. The online change phase failed on this IMS because of the state of the specified resource. Many of the resource states indicate work is in progress for resources to be changed or deleted by online change. Online change commit detected a resource in a state that results in commit failure. This is the first resource for which an error is detected. IMS is returned to an online change prepare state. Issue a `/DISPLAY MODIFY` command to display the work in progress for resources to be changed or deleted by online change. Resolve the work in progress, then attempt the `INITIATE OLC PHASE(COMMIT)` command again.	Resource type (8 char), resource name (8 char), error detail (16 char). The resource name is the name of the resource as it is defined to IMS, such as the database name, the program name, or the transaction name. The resource type can be DATABASE, DMB, FORMAT, PROGRAM, PSB, RTCODE, or TRAN.
B1	Resource state error	The error detail can be one of the following: • ACTIVE The routing code rtcode is active. • AREA OPEN An area associated with the FP DEDB is open. • AREA STARTED An area associated with the FP DEDB is started. • CONVERSATION The transaction to be changed or deleted or the transaction referencing a program or database to be changed or deleted is in conversation. • DBD ACTIVE A `/DBDUMP` command is active for the database. • DBR ACTIVE A `/DBRECOVERY` command is active for the database. • DBR NEEDED A `/DBRECOVERY` command is needed for the database resource. Online change was initiated to either change or delete one or more HALDBs which have not been taken off-line with a `/DBR` command. • DEP REGS ACTIVE Dependent regions using the FP DEDB are active.

Table 70. Completion Codes for the INITIATE OLC Command (continued)

Completion Code	Meaning	ERROR TEXT (upper case)
B1	Resource state error (continued)	Error detail (continued): • DYNAMIC OR CPIC The transaction being added already exists and is not dynamic or CPIC. This is an internal error that should not occur. • INTERNAL ERROR The resource control block is not found in the active IMS, or the resource control block already exists. These are internal errors that should not occur. • IN USE The resource is in use. A transaction has queueing in progress (either terminal input program-to-program switch). A database is in use. • PSB SCHEDULED A program referencing a database to be changed or deleted is scheduled. • QUEUEING Messages are queued to the transaction or program. • SCHEDULED The named resource (a transaction or program to be changed or deleted, or a program referencing a database to be changed or deleted) is scheduled. • SUSPENDED The transaction to be changed or deleted is on the suspend queue.

Table 70. Completion Codes for the INITIATE OLC Command (continued)

Completion Code	Meaning	ERROR TEXT (upper case)
B2	IMS state error.	IMS state error (32 char). The IMS state can be one of the following: • ABENDED This IMS abended since the last successful online change. Online change is not permitted if any IMS abended since the last online change, unless the FRCABND option is specified on prepare. If the online change phase was rejected, issue `TERMINATE OLC` to abort the online change, correct the problem, and try the online change again. If an online change specified with the FRCABND option succeeds, this IMS may have to coldstart. • CHECKPOINT IN PROGRESS This IMS has checkpoint in progress. • NOT-REACHABLE The online change phase is rejected because this IMS is NOT-REACHABLE. The SCI on the OS image where this IMS is active is down. Restart the SCI and re-issue the `INITIATE OLC` or `TERMINATE OLC` command. • OLC ALREADY COMMITTED The online change phase is rejected because online change is already committed. All IMSs have completed commit phase 1 and the OLCSTAT data set was updated. • OLC NOT IN PROGRESS The online change phase is rejected because this IMS is not in an online change state.

Table 70. Completion Codes for the INITIATE OLC Command (continued)

Completion Code	Meaning	ERROR TEXT (upper case)
B2	IMS state error (continued).	IMS state error (32 char). The IMS state can be one of the following: • OLC PHASE IN PROGRESS The online change phase is rejected because this IMS has an online change phase already in progress. • ORS IN PROGRESS The Online change command is rejected because the online recovery service is already in progress. • RESOLVE DB INDOUBTS This IMS has DB indoubts. You must resolve the DB indoubts either by reconnecting the coordinator controller and IMS or by using an operator command. • RESTART IN PROGRESS This IMS has restart in progress. • RESTART NOT COMPLETE This IMS initialized before the online change was initiated, but has not completed restart. The online prepare or abort phase is rejected as long as this IMS is in this state. Cancel this IMS, then abort the online change before attempting the online change prepare phase again. • SHUTDOWN This IMS shut down normally since the last successful online change. Online change is not permitted if any IMS shut down normally since the last online change, unless the FRCNRML option is specified on prepare. If the online change phase was rejected, issue `TERMINATE OLC` to abort the online change, correct the problem, and try the online change again. If an online change specified with the FRCNRML option succeeds, this IMS may have to cold start.

INITIATE OLC Error Handling

The `INITIATE OLC` command can result in errors that leave one or more of the IMSs in the IMSplex in various online change states. Issue the `QUERY MEMBER` command and the `QUERY OLC` command to help you determine whether to terminate the online change or try the `INITIATE OLC` command again.

Before attempting online change, issue the `QUERY OLC LIBRARY(OLCSTAT) SHOW(MODID)` command to get the current modify id. If the `INITIATE OLC` command fails, issue the `QUERY OLC LIBRARY(OLCSTAT) SHOW(MODID)` command again, to see if the modify id is the same. If the modify id increased by 1, the online change is considered to be successfully completed.

If the INITIATE OLC command fails, issue the QUERY MEMBER TYPE(IMS) SHOW(STATUS) command to display the online change state of all the IMSs in the IMSplex. Evaluate the QUERY MEMBER TYPE(IMS) SHOW(STATUS) output to help you to determine what to do:

- None of the IMSs in an online change state

  If the INITIATE OLC PHASE(PREPARE) command failed on all of the IMSs in the IMSplex, none of them are in an online change state. This QUERY OLC LIBRARY(OLCSTAT) SHOW(MODID) command output shows the same modify id as the QUERY OLC LIBRARY(OLCSTAT) SHOW(MODID) command issued before the INITIATE OLC PHASE(PREPARE) command. If you want to attempt online change again, issue another INITIATE OLC PHASE(PREPARE) command.

  If an INITIATE OLC PHASE(COMMIT) command successfully completes commit phase 3 on all the IMSs but fails due to an error such as a command timeout, then no IMS will remain in an online change state. The QUERY MEMBER TYPE(IMS) SHOW(STATUS) command output shows no IMS in an online change state, so no further action is needed. The online change successfully completed.

  If an INITIATE OLC PHASE(PREPARE) command failed for all the IMSs in the IMSplex, but RM still has information about the process, then the online change needs to be terminated. This is a very rare case where the command master initiates the online change process with RM, an error occurs, or the online change prepare phase fails on all the IMSs, and the command master in unable to terminate the online change process with RM. RM issues a CSLR2200 message but no CSLR2201 message. The QUERY MEMBER TYPE(IMS) SHOW(STATUS) command output shows no IMS in an online change state. If you try to initiate an online change again with another INITIATE OLC PHASE(PREPARE) command, it is rejected. You must terminate the online change with a TERMINATE OLC command.

- Some of the IMSs in a prepare complete state

  If the INITIATE OLC PHASE(PREPARE) command fails and some of the IMSs show a status of online change prepare complete (OLCPREPC), then the prepare phase succeeded on some of the IMSs. You must terminate the online change with the TERMINATE OLC command. The IMSs that have a status of OLCPREPC remain in an online change prepare complete state until you abort the online change.

- All IMSs in a prepare complete state

  If the INITIATE OLC PHASE(PREPARE) or INITIATE OLC PHASE(COMMIT) command fails and all of the IMSs are in a prepare complete state (OLCPREPC) and the master shows a global status of online change prepare complete (OLCPREPC), then the prepare succeeded. You can issue INITIATE OLC PHASE(COMMIT) if you want to proceed with the online change. You can issue TERMINATE OLC if you want to abort the online change.

  If the INITIATE OLC PHASE(PREPARE) command flavor did not apply to any IMS in the IMSplex, the command is rejected, but all of the IMSs show a status of OLCPREPC. For example, if the IMSplex consists of only DBCTL subsystems, then the INITIATE OLC PHASE(PREPARE) TYPE(FMTLIB) command does not apply to any of the IMSs and is rejected. However, all of the IMSs will show a status of OLCPREPC, indicating they are participating in the online change. If this situation occurs, you must terminate the online change. INITIATE OLC PHASE(COMMIT) will be rejected, since the online change applies to no IMS in the IMSplex.

- Some IMSs in prepare complete and commit phase 1 complete state

  If an INITIATE OLC PHASE(COMMIT) command fails during commit phase 1, some of the IMSs may be in a prepare complete state (OLCPREPC) and some of the IMSs may be in a commit phase 1 complete state (OLCCMT1C). The commit

phase failed before the master updated the OLCSTAT data set, so the online change is not committed. You may correct the problem and try the commit again with the INITIATE OLC PHASE(COMMIT) command. Or, you may decide to abort the online change with the TERMINATE OLC command.

The IMSs that are in an online change state remain in an online change state until you abort the online change or commit the online change.

- All IMSs in a commit phase 1 complete state

  If an INITIATE OLC PHASE(COMMIT) command fails after commit phase 1 is completed, all of the IMSs are in a commit phase 1 complete state (OLCCMT1C).

  If the error occurs before the master updates the OLCSTAT data set, then the online change is not yet successful. You may abort the online change with the TERMINATE OLC command or attempt the commit again with another INITIATE OLC PHASE(COMMIT) command.

  If the error occurs after the master updates the OLCSTAT data set, then the online change is successful. You may not abort the online change. You must attempt the commit again with another INITIATE OLC PHASE(COMMIT) command to finish up the online change. The INITIATE OLC PHASE(COMMIT) command finishes the online change by completing commit phase 2.

  The IMSs that are in an online change state remain in an online change state until you abort the online change or commit the online change.

  You can determine if the OLCSTAT data set has been updated by checking the modify id. Issue the QUERY OLC LIBRARY(OLCSTAT) SHOW(MODID) command. Check if the modify id returned is different from the modify id returned by the INITIATE OLC PHASE(PREPARE) command, or the modify id returned by a QUERY OLC LIBRARY(OLCSTAT) SHOW(MODID) command issued before the INITIATE OLC PHASE(COMMIT) command.

- Some IMSs in commit phase 1 complete state and some in commit phase 2 complete state

  If an INITIATE OLC PHASE(COMMIT) command fails during commit phase 2, some of the IMSs may be in a commit phase 1 complete state (OLCCMT1C) and some may have completed commit phase 2 and be in a commit phase2 complete state (OLCCMT2C). The INITIATE OLC PHASE(COMMIT) command failed after commit phase 1 completed on all of the IMSs and the master updated the OLCSTAT data set. Once the OLCSTAT data set is updated, the online change is considered to be successful and cannot be aborted. You must finish the online change commit phase 2 by issuing the INITIATE OLC PHASE(COMMIT) command again. The INITIATE OLC PHASE(COMMIT) command finishes the online change by completing commit phase 2.

  The IMSs that are in an online change state remain in an online change state until you finish the online change with an INITIATE OLC PHASE(COMMIT) command.

- All IMSs in commit phase 2 complete state

  If an INITIATE OLC PHASE(COMMIT) command fails during commit phase 2 or commit phase 3, all of the IMSs may end up in a commit phase 2 complete state (OLCCMT2C). The INITIATE OLC PHASE(COMMIT) command failed after commit phase 1 completed on all of the IMSs and the master updated the OLCSTAT data set. Once the OLCSTAT data set is updated, the online change is considered to be successful and can not be aborted. you must finish the online change commit phase 2 and commit phase 3 by issuing the INITIATE OLC PHASE(COMMIT) command again.

The IMSs that are in an online change state remain in an online change state until you finish the online change with an INITIATE OLC PHASE(COMMIT) command.

- Some IMSs in commit phase 2 complete state and some not in online change state

  If an INITIATE OLC PHASE(COMMIT) command fails during commit phase 3, some of the IMSs may be in a commit phase 2 complete state (OLCCMT2C) and some may have completed commit phase 3 and no longer be in an online change state. The INITIATE OLC PHASE(COMMIT) command failed after commit phase 1 completed on all of the IMSs and master updated the OLCSTAT data set. Once the OLCSTAT data set is updated, the online change is considered to be successful and can not be aborted. You must finish the online change commit phase 3 by issuing the INITIATE OLC PHASE(COMMIT) command again, routing it to an IMS that is still in a commit phase 2 complete state. The INITIATE OLC PHASE(COMMIT) command finishes the online change by completing commit phase 3.

  The IMSs that are in an online change state remain in an online change state until you finish the online change with an INITIATE OLC PHASE(COMMIT) command.

Errors unique to the processing of this command are returned as a completion code. A completion code is returned for an IMS participating in the online change phase. See Table 70 on page 398 for a listing of the completion codes that may be returned on an INITIATE OLC command.

Examples for INITIATE OLC Command

The following are examples of the INITIATE OLC command.

Example 1 for INITIATE OLC Command

TSO SPOC input:

```
INITIATE OLC PHASE(PREPARE) TYPE(MODBLKS,ACBLIB)
```

TSO SPOC output:

```
MbrName  Member     CC ACBLIB FMTLIB MODBLKS    ModId
IMS3     IMS2        0
IMS3     IMS3        0      B      A       B        1
IMS3     SYS3        0
```

OM API input:

```
CMD (INITIATE OLC PHASE(PREPARE) TYPE(MODBLKS,ACBLIB))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2003.163 15:40:53.336327</statime>
<stotime>2003.163 15:41:08.282146</stotime>
<staseq>B7C4AC11FED070C6</staseq>
<stoseq>B7C4AC203FB220C1</stoseq>
<rqsttkn1>USRT011 10084053</rqsttkn1>
<rc>0200000C</rc>
<rsn>00003000</rsn>
</ctl>
<cmderr>
<mbr name="IMS2    ">
```

```
<typ>IMS     </typ>
<styp>DBDC  </styp>
<rc>02000004</rc>
<rsn>00001008</rsn>
</mbr>
<mbr name="SYS3    ">
<typ>IMS     </typ>
<styp>DBDC   </styp>
<rc>02000004</rc>
<rsn>00001008</rsn>
</mbr>
</cmderr>
<cmd>
<master>IMS3    </master>
<userid>USRT011 </userid>
<verb>INIT</verb>
<kwd>OLC            </kwd>
<input>INITIATE OLC PHASE(PREPARE) TYPE(MODBLKS,ACBLIB)</input>
</cmd>
<cmdrsphdr>
<hdr slbl="MBR"    llbl="MbrName" scope="LCL" sort="n" key="0" scroll="yes" len="8"
 dtype="CHAR" align="left" />
<hdr slbl="IMSMBR" llbl="Member"  scope="LCL" sort="a" key="1" scroll="no"  len="8"
 dtype="CHAR" align="left" />
<hdr slbl="CC"     llbl="CC"      scope="LCL" sort="n" key="0" scroll="yes" len="4"
 dtype="INT"  align="right" />
<hdr slbl="ACBL"   llbl="ACBLIB"  scope="GBL" sort="n" key="0" scroll="yes" len="1"
 dtype="CHAR" align="right" />
<hdr slbl="FMTL"   llbl="FMTLIB"  scope="GBL" sort="n" key="0" scroll="yes" len="1"
 dtype="CHAR" align="right" />
<hdr slbl="MODB"   llbl="MODBLKS" scope="GBL" sort="n" key="0" scroll="yes" len="1"
 dtype="CHAR" align="right"/>
<hdr slbl="MODI"   llbl="ModId"   scope="GBL" sort="n" key="0" scroll="yes" len="8"
 dtype="CHAR" align="right"/>
</cmdrsphdr>
<cmdrspdata>
<rsp>MBR(IMS3    ) IMSMBR(SYS3    ) CC(   0) </rsp>
<rsp>MBR(IMS3    ) IMSMBR(IMS2    ) CC(   0) </rsp>
<rsp>MBR(IMS3    ) IMSMBR(IMS3    ) CC(   0) ACBL(B) FMTL(A) MODB(B) MODI(    1)
</rsp>
</cmdrspdata>
</imsout>
```

Explanation: The IMSplex consists of three IMSs -- SYS3, IMS2, and IMS3. All three of the IMSs complete the online change prepare phase successfully for the MODBLKS library. IMS3 is the master of the prepare phase. The current online change libraries are ACBLIBB, FMTLIBA, and MODBLKSB. The current modify id is 1.

Example 2 for INITIATE OLC Command

TSO SPOC input:

```
INITIATE OLC PHASE(COMMIT)
```

TSO SPOC output:

```
MbrName  Member     CC ACBLIB FMTLIB MODBLKS    ModId
IMS3     IMS2        0
IMS3     IMS3        0      B      A       A        2
IMS3     SYS3        0
```

OM API input:

```
CMD (INITIATE OLC PHASE(COMMIT))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2002.163 16:26:14.660563</statime>
<stotime>2002.16316:26:22.293891</stotime>
<staseq>B7C4B635409D30C6</staseq>
<stoseq>B7C4B63C88383347</stoseq>
<rqsttkn1>USRT011 10092614</rqsttkn1>
<rc>0200000C</rc>
<rsn>00003000</rsn>
</ctl>
<cmderr>
<mbr name="IMS2    ">
<typ>IMS     </typ>
<styp>DBDC   </styp>
<rc>02000004</rc>
<rsn>00001008</rsn>
</mbr>
<mbr name="SYS3    ">
<typ>IMS     </typ>
<styp>DBDC    </styp>
<rc>02000004</rc>
<rsn>00001008</rsn>
</mbr>
</cmderr>
<cmd>
<master>IMS3    </master>
<userid>USRT011 </userid>
<verb>INIT</verb>
<kwd>OLC            </kwd>
<input>INITIATE OLC PHASE(COMMIT) </input>
</cmd>
<cmdrsphdr>
<hdr slbl="MBR"    llbl="MbrName" scope="LCL" sort="n" key="0" scroll="yes" len="8"
 dtype="CHAR" align="left" />
<hdr slbl="IMSMBR" llbl="Member"  scope="LCL" sort="a" key="1" scroll="no"  len="8"
 dtype="CHAR" align="left" />
<hdr slbl="CC"     llbl="CC"      scope="LCL" sort="n" key="0" scroll="yes" len="4"
 dtype="INT"  align="right" />
<hdr slbl="ACBL"   llbl="ACBLIB"  scope="GBL" sort="n" key="0" scroll="yes" len="1"
 dtype="CHAR" align="right" />
<hdr slbl="FMTL"   llbl="FMTLIB"  scope="GBL" sort="n" key="0" scroll="yes" len="1"
 dtype="CHAR" align="right" />
<hdr slbl="MODB"   llbl="MODBLKS" scope="GBL" sort="n" key="0" scroll="yes" len="1"
 dtype="CHAR" align="right"/>
<hdr slbl="MODI"   llbl="ModId"   scope="GBL" sort="n" key="0" scroll="yes" len="8"
 dtype="CHAR" align="right" />
</cmdrsphdr>
<cmdrspdata>
<rsp>MBR(IMS3    ) IMSMBR(SYS3    ) CC(   0) </rsp>
<rsp>MBR(IMS3    ) IMSMBR(IMS2    ) CC(   0) </rsp>
<rsp>MBR(IMS3    ) IMSMBR(IMS3    ) CC(   0) ACBL(B) FMTL(A) MODB(A) MODI(      2)
</rsp>
</cmdrspdata>
</imsout>
```

Explanation: A global online change commit is initiated and completes successfully.

Chapter 32. /LOG

Format

Environments

Table 71 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command can be issued.

Table 71. Valid Environments for the /LOG Command

Command	DB/DC	DBCTL	DCCTL
/LOG	X	X	X

Usage

/LOG writes a single-segment message to the IMS system log. Any text can be logged, including transaction codes or command streams. An X'02' log record is produced and contains the text. This command applies only to the currently entered message segment and does not establish a continuing operational mode.

text is the alphanumeric character message to be logged.

Example for /LOG Command

Entry ET:

```
/LOG TODAY IS MONDAY
```

Response ET:

```
DFS058I  LOG COMMAND COMPLETED
```

Explanation: An X'02' log record containing the text

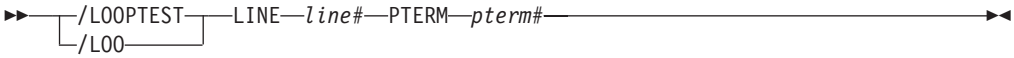
```
TODAY IS MONDAY
```

is written to the IMS system log.

© Copyright IBM Corp. 1974, 2003

Chapter 33. /LOOPTEST

Format



Environments and Keywords

Table 72 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 72. Valid Environments for the /LOOPTEST Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/LOOPTEST	X		X
LINE	X		X
PTERM	X		X

Usage

/LOOPTEST is a multisegment command that creates an output write loop that tests for output errors. It causes a single-segment message to be repeatedly transmitted to the terminal being tested.

Multisegment input is combined into a single segment for output transmission. For the terminal in looptest mode to receive the complete message, the cumulative output segment must not exceed the terminal's output size restrictions.

/LOOPTEST requires an EOM indication to denote end-of-message; an EOS indication must be included for all segments that precede the last segment. See "Multisegment Command Input" on page 8 for more detail on using EOS and EOM.

/LOOPTEST causes all I/O error detection and notification procedures within IMS to be bypassed. Depending on the types of errors involved, some write loops might result in high processor utilization.

The /END command terminates looptest mode.

LINE, PTERM
Specifies the communication line or physical terminal to be placed into looptest mode. If the terminal in looptest mode is attached to a multipoint line, the entire line must be stopped and idle before the /LOOPTEST command is allowed. /LOOPTEST is not allowed on VTAM-attached terminals, switched lines, or lines for System/3 and System/7.

If the device specified is not powered on and ready, intervention-required messages will be sent to the master terminal.

The /DISPLAY LINE PTERM command identifies a terminal in looptest mode.

Example for /LOOPTEST Command

Entry ET:

© Copyright IBM Corp. 1974, 2003

```
/LOOPTEST LINE 5 PTERM 1 (EOS)
DATA TEST (EOS)
12345 (EOM)
```

Response ET:

```
DFS058I  LOOPTEST COMMAND COMPLETED
```

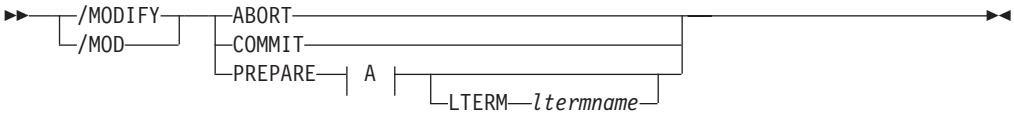
Response RT:

```
DATA TEST 12345
DATA TEST 12345
and so forth.
```

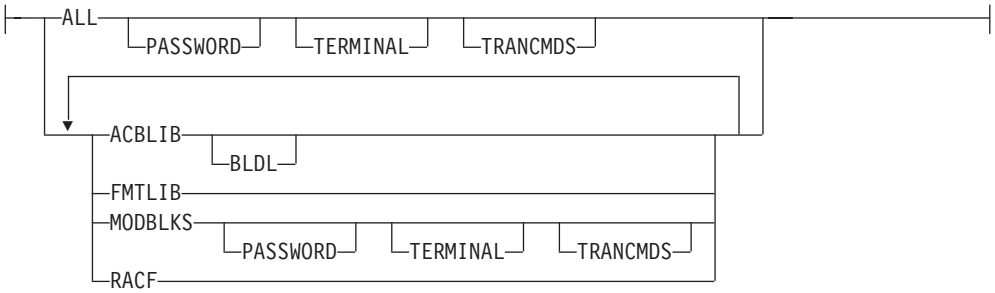
Explanation: DATA TEST 12345 is repetitively sent to PTERM 1 on LINE 5.

Chapter 34. /MODIFY

Format



A:



Environments and Keywords

Table 73 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 73. Valid Environments for the /MODIFY Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/MODIFY	X	X	X
ABORT	X	X	X
COMMIT	X	X	X
LTERM	X		X
PASSWORD	X	X	X
PREPARE	X	X	X
TERMINAL	X		X
TRANCMDS	X		X

Usage

/MODIFY controls the modification of IMS resources online. It switches the online environment from the active ACBLIB, FORMAT, MATRIX, and MODBLKS libraries to the inactive libraries containing the new or changed resource descriptions. See *IMS Version 9: Administration Guide: System* and *IMS Version 9: Installation Volume 1: Installation Verification* on how to create the inactive libraries before using this command. See *IMS Version 9: Utilities Reference: System* for information on using the Online Change utility. The descriptions of the resources and parameters for the /MODIFY command are shown in Table 74 on page 418

© Copyright IBM Corp. 1974, 2003

Table 74. Resources and Parameters Available to the /MODIFY Command

Parameter	Library	Description
ACBLIB	IMS.ACBLIBx [1]	Contains database and program descriptors such as DMBs and PSBs.
FMTLIB	IMS.FORMATx [1]	Contains MFS maps produced by the MFS language utility and service utility. The FMTLIB parameter is not valid with DBCTL.
MODBLKS	IMS.MODBLKSx, [1] IMS.MATRIXx	Contains a subset of control blocks produced by IMS system definition and security definitions.
RACF		RACF in-storage profiles. The RACF parameter is not valid with DBCTL.

Note:

1. × = A or B
2. Use the SETROPTS RACLIST(CIMS) REFRESH command to refresh RACF updates.

The correct sequence for /MODIFY commands is PREPARE followed by COMMIT. If the online change is not to be completed, the /MODIFY ABORT command is used. The /MODIFY PREPARE command specifies which resources are to be changed or deleted. Depending on the parameters entered, the system will initiate quiescing of appropriate resources. COMMIT initiates the changes entered in a preceding PREPARE. If a previously entered /MODIFY PREPARE command is to be canceled, ABORT must be entered.

An IMS automated operator program can issue the /MODIFY PREPARE and /MODIFY ABORT commands, but it cannot issue the /MODIFY COMMIT command. Thus, you can automatically prepare or abort online change.

If this command is used to delete a database, IMS closes the database and makes it unavailable to programs. Also, if the database is authorized for DBRC, /MODIFY unauthorizes it.

Fast Path DEDBs are supported by the /MODIFY command, but Fast Path MSDBs are not supported.

In an IMSplex, /MODIFY is not supported if coordinated online change is enabled.

ABORT

Resets the status that was set by the /MODIFY PREPARE command after a /MODIFY PREPARE, or can be used if the /MODIFY COMMIT was not successful and the operator chooses not to attempt the online change at that time.

COMMIT

Brings all the newly defined resources online, updates the changed resources, and invalidates the deleted resources.

If ALL was specified on the /MODIFY PREPARE command, the ACBLIB, FMTLIB, and MODBLKS changes must be successful or the /MODIFY COMMIT will be rejected. When COMMIT is successful, the modifications will persist across all IMS restarts. If FMTLIB is entered, most new and in-progress requests will be queued for processing after the /MODIFY COMMIT command is completed.

If MODBLKS or ACBLIB is specified, work in progress in the system for changed or deleted ACBLIB and MODBLKS resources must be completed, or the /MODIFY command is rejected.

/DISPLAY MODIFY is used to display the work that was in progress prior to the entering of /MODIFY COMMIT.

IMS will prevent messages entered from terminals from being queued to transactions, but program-to-program switches from applications can still occur. Therefore, a /DISPLAY MODIFY command can indicate that no work is pending for the resources to be modified or deleted, but the /MODIFY COMMIT might be rejected. If this occurs, the new work must be allowed to complete and then the /MODIFY COMMIT command must be reentered.

/MODIFY COMMIT is rejected if:

- A /MODIFY PREPARE command was not previously entered.
- Transactions to be deleted have messages queued.
- Transactions to be changed by the following system definition keywords have messages queued:
  - MSGTYPE
  - INQUIRY
  - FPATH
  - EDIT
  - SPA
- Transactions to be changed by keywords not shown in the previous bullet have messages queued, and the transactions have not been the object of a /STOP or /PSTOP command.
- Transactions with access to programs or databases to be changed or deleted are prevented from updating, and the transactions have not been the object of a /STOP or /PSTOP command. When a transaction shows a status of USTOP during a /DISPLAY TRANSACTION, the /STOP command might need to be entered, because the /PSTOP command might not allow the /MODIFY command to complete the transaction.
- Programs or databases to be changed or deleted are scheduled.
- Any program (PSB) that is currently scheduled has a program, database, or transaction change or deletion. This includes Wait for Input (WFI) and Fast Path transactions. All WFI and Fast Path regions that reference changed or deleted routing codes, programs, or databases must be stopped before entering the /MODIFY COMMIT.
- An I/O error occurs while reading the inactive MFS FORMAT library directory or the resident index ($$IMSDIR) member.
- A /START DATABASE command is in progress for any database that is changed or deleted.
- A /DBDUMP or /DBRECOVERY command is in progress for any database that is changed or deleted.

Operator action might be required to prevent the preceding conditions. For example, if a program or database is scheduled, the operator should either wait until the program or database is finished before the /MODIFY COMMIT command is entered, or the operator should issue /STOP or /PSTOP for the associated transaction.

If IMS is in an IMSplex and Resource Manager (RM) is using a resource structure, the transactions being added by the online change are registered to RM if they are not already registered to RM. If any error occurs during the CSLRMUPD request to register the transactions, the /MODIFY or INITIATE OLC command will fail and IMS will be in a COMMIT2F state.

If a /MODIFY COMMIT command results in the COMMIT2F state, the command is rejected with a DFS178 message. A DFS3308 message is sent to the system console with the CSLRMUPD return/reason code. The IMS remains in a COMMIT2F state and the /MODIFY cannot be committed or aborted. The IMS must be cancelled and warm started as the online change is not complete.

PREPARE

/MODIFY PREPARE is not recovered across an emergency restart and must be reentered after restart if the /MODIFY COMMIT did not complete prior to an IMS failure.

The parameters in Table 75 are valid with /MODIFY PREPARE. The environments in which the parameters are valid are also shown in the following table.

Table 75. /MODIFY PREPARE Parameters Environments

Parameter	DB/DC	DBCTL	DCCTL
ACBLIB	X	X	X
ALL	X	X[1]	X
BLDL	X	X	X
FMTLIB	X		X
LTERM	X		X
MODBLKS	X	X	X
PASSWORD	X	X	X
TERMINAL	X		X
TRANCMDS	X		X
RACF	X		X

Note:

1. Only applies to ACBLIB, MODBLKS, and BLDL.

ACBLIB

Quiesces the transactions to be changed or deleted and transactions that access databases or programs to be changed or deleted. ACBLIB members that are not found are displayed as CHANGED because a DDIR or PDIR exists in MODBLKS. Additions or deletions to ACBLIB require the MODBLKS parameter.

ALL

Indicates that /MODIFY PREPARE is to be performed for ACBLIB, FMTLIB, RACF, MODBLKS and BLDL.

BLDL

Causes information message DFS3498I to be issued for each PSB or DMB not found in ACBLIB. This message does not affect the success of the command.

FMTLIB

Indicates that there will be no restrictions on the use of MFS format blocks until the /MODIFY COMMIT command is entered.

LTERM ltermname

Specifies a logical terminal to receive lists of added, changed, or deleted resources.

MODBLKS

Causes IMS to prepare to bring the changes or deletions defined by the

new system definition to IMS. Using MODBLKS also quiesces the transactions to be changed or deleted and transactions that access databases or programs to be changed or deleted.

PASSWORD
Is an optional keyword that can be specified in any combination with ALL or MODBLKS to add, but not delete, specific types of security definitions if they had not existed before. If ALL or MODBLKS is specified, the application of new security definitions will always be attempted based upon the prevailing security options in effect from the last IMS initialization.

Related Reading: See *IMS Version 9: Administration Guide: System* for more information on security definition.

RACF
Use of the RACF data space invalidates the IMS online change support for RACF with the `/MODIFY` command. The IMS online change support is still valid, though, when the RACF data space is not being used.

The message `DFS3432 RACF PARAMETER INVALID IF RACF DATA SPACE IS USED` is issued if the RACF parameter is used on the `/MODIFY PREPARE` command when the RACF data space is being used. You can use the RACF command `SETROPTS RACLIST (classname) REFRESH` to refresh the RACF resource profiles in the RACF data space without requiring the IMS applications to suspend work.

TERMINAL
Is an optional keyword that can be specified in any combination with ALL or MODBLKS to add, but not delete, specific types of security definitions if they had not existed before. If ALL or MODBLKS is specified, the application of new security definitions will always be attempted based upon the prevailing security options in effect from the last IMS initialization.

Related Reading: See *IMS Version 9: Administration Guide: System* for more information on security definition.

TRANCMDS
Is an optional keyword that can be specified in any combination with ALL or MODBLKS to add, but not delete, specific types of security definitions if they had not existed before. If ALL or MODBLKS is specified, the application of new security definitions will always be attempted based upon the prevailing security options in effect from the last IMS initialization.

Related Reading: See *IMS Version 9: Administration Guide: System* for more information on security definition.

Examples

Example 1 for /MODIFY Command

This is an example of a successful PREPARE, COMMIT sequence. First, the command `/MODIFY PREPARE ALL` is issued, as shown below:

Entry ET:

```
/MODIFY PREPARE ALL
```

Response ET:

```
DFS3499I ACTIVE DDNAMES MODBLKSA IMSACBA FORMATA MODSTAT ID: 103
```

Explanation: The modifications are brought online to IMS during a subsequent successful /MODIFY COMMIT command. The ddnames of the active data sets and the current MODSTAT identifier are shown. This message is the normal response.

Example 2 for /MODIFY Command

This is an example of a prepare/commit/abort sequence.

Entry ET:

```
/MODIFY COMMIT
```

Response ET:

```
DFS3499I ACTIVE DDNAMES: MODBLKSB IMSACBB FORMATB MODSTAT ID: 104
```

Explanation: Successful COMMIT. ID is incremented and DDNAMES changed. This message is the normal response.

After the response is received from the second part of the successful PREPARE, COMMIT sequence, the command shown in the example below is issued to conclude the PREPARE, COMMIT sequence:

Entry ET:

```
/MODIFY PREPARE ACBLIB MODBLKS TERMINAL
```

Response ET:

```
DFS3499I ACTIVE DDNAMES MODBLKSB IMSACBB FORMATB MODSTAT ID: 104
```

Explanation: The modifications will be brought online to IMS during a subsequent successful /MODIFY COMMIT command. This message is the normal response.

Entry ET:

```
/MODIFY COMMIT
```

Response ET:

```
DFS3452I WORK IN PROGRESS FOR RESOURCES TO BE CHANGED/DELETED
```

Explanation: The /MODIFY COMMIT was rejected because changed/deleted resources have not been quiesced.

Entry ET:

```
/MODIFY ABORT
```

Response ET:

```
DFS058I  MODIFY ABORT COMMAND COMPLETED
```

Explanation: Status set by the /MODIFY PREPARE command is reset and the online change does not occur.

Example 3 for /MODIFY Command

This example shows the use of the LTERM keyword when adding, changing, or deleting Fast Path DEDBs or areas.

Entry ET:

```
/MODIFY PREPARE ALL LTERM WTOR
```

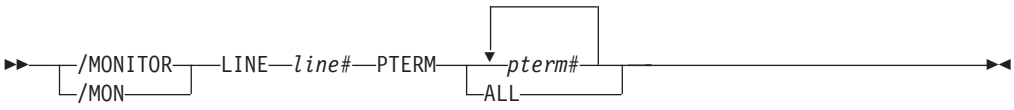
Response ET:

```
DFS3499I ACTIVE DDNAMES: MODBLKSA IMSACBA  FORMATA  MODSTAT ID:    1
DFS3430I THE FOLLOWING MODIFICATIONS WILL OCCUR DURING /MODIFY ...
COMMIT:
DFS3430I DATABASE CALENDER      ADDED
DFS3430I DATABASE NEWDD01       ADDED
DFS3430I DMB      CALENDER      ADDED
DFS3430I DMB      NEWDD01       ADDED
DFS3430I DMB      DEDBJN23      CHANGED
DFS3430I   AREA   DB23AR4       DELETED
DFS3430I   AREA   DB23AR5       DELETED
DFS3430I   AREA   DB23AR6       DELETED
DFS3430I   AREA   DB23AR7       DELETED
DFS3430I DMB      DEDBJN24      CHANGED
DFS3430I   AREA   DB24ANU1      ADDED
DFS3430I   AREA   DB24ANU2      ADDED
DFS3430I   AREA   DB24ANU3      ADDED
DFS3430I PSB      BMP255        CHANGED
DFS3430I PSB      DDLTFPE4      CHANGED
DFS3430I PSB      DDLTJN21      CHANGED
DFS3430I PSB      DDLTRN20      CHANGED
DFS3430I TRAN     TRAN255       CHANGED
DFS3430I TRAN     TXCDRN20      CHANGED
DFS3430I TRAN     TXCD255       CHANGED
DFS3430I TRAN     TXCD255W      CHANGED
DFS3430I SECURITY PASSWORD      NONE
DFS3430I SECURITY TERMINAL      NONE
DFS3430I SECURITY TRANCMDS      NONE
DFS3430I SECURITY SIGNON        NONE
DFS3430I SECURITY AGT           NONE
DFS3430I END OF MODIFICATIONS
```

Explanation: The modifications will be brought online to IMS during a subsequent successful `/MODIFY COMMIT` command. These messages are the normal response.

Chapter 35. /MONITOR

Format



Environments and Keywords

Table 76 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 76. Valid Environments for the /MONITOR Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/MONITOR	X		X
LINE	X		X
PTERM	X		X

Usage

/MONITOR stops output from being sent to programmable remote stations (System/3 or System/7). Input and output message queuing are allowed to continue.

/MONITOR can be used to reset conditions previously established by the /START, /RSTART, /STOP, /PSTOP, or /PURGE command.

LINE, PTERM
Specifies the communication line or physical terminal for which output is being stopped.

Example for /MONITOR Command

Entry ET:

```
/MONITOR LINE 4 PTERM 3
```

Response ET:

```
DFS058I  MONITOR COMMAND COMPLETED
```

Response RT:

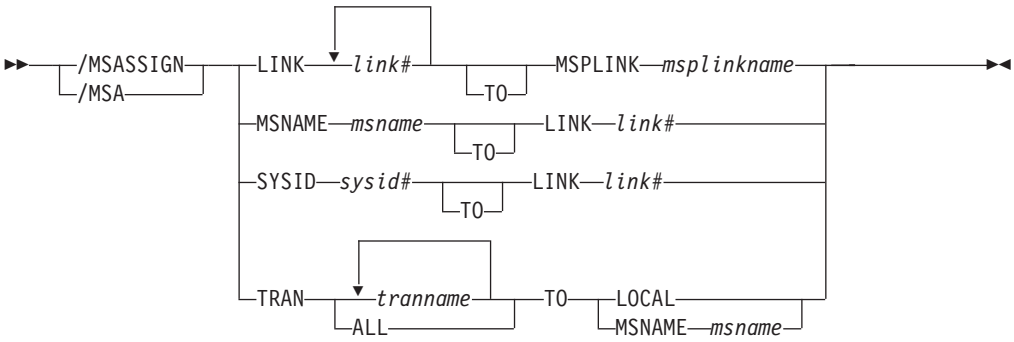
```
DFS059I  TERMINAL MONITORED
```

Explanation: No further output is allowed to PTERM 3 on LINE 4.

© Copyright IBM Corp. 1974, 2003

Chapter 36. /MSASSIGN

Format



Environments and Keywords

Table 77 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 77. Valid Environments for the /MSASSIGN Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/MSASSIGN	X		X
LINK	X		X
LOCAL	X		X
MSNAME	X		X
MSPLINK	X		X
SYSID	X		X
TRAN	X		X

Usage

The /MSASSIGN command alters the assignments of the following multisystem resources: LINK, MSPLINK, MSNAME, and SYSID. These assignments include logical link to physical link, remote system identification (SYSID) to logical link, logical link path (MSNAME) to logical link, and transaction to a local or remote system.

All changes made by an /MSASSIGN command remain until changed with another /MSASSIGN command or an IMS cold start is performed. After the /MSASSIGN command is used, /MSVERIFY should be used to ensure that the assignment produced a valid configuration.

LINK

Specifies the logical link in a multiple system configuration.

Logical links can be assigned to physical links for input/output purposes. Only one logical link at a time can be assigned to a physical link (except for VTAM MSPLINKs, which can have multiple logical links assigned to them). The logical link must be stopped and idle for this assignment. Systems can only

© Copyright IBM Corp. 1974, 2003

communicate with each other across a logical link established through specification of the same partner ID in both systems.

This type of assignment must be synchronized between the master terminal operators of the two systems. If a connection becomes inoperable while a transmission is in progress, this type of assignment allows switching to an alternate connection and proceeding with the unfinished transmission.

LOCAL
Specifies the local system. that is the subsystem in which the command is issued. The command `/MSASSIGN TRANSACTION p1 TO LOCAL` assigns transaction p1 to the local system.

MSNAME
Specifies the logical link path in a multiple systems configuration.

MSPLINK
Specifies the physical link in a multiple subsystem configuration.

The command `/MSASSIGN LINK p1 MSPLINK p2` assigns the logical link p1 to the physical link p2 for input and output purposes.

SYSID
Specifies the system identification of a remote system in a multiple system configuration. Remote SYSIDs can be assigned to logical links for output purposes. More than one SYSID (=MSNAME) can be assigned to one logical link. A remote SYSID (=MSNAME) cannot be assigned to another logical link if a message is currently transmitting on the connection corresponding to this SYSID. Therefore, the logical link must be stopped and idle for this assignment.

TRANSACTION
Specifies the transaction to be assigned to either the local or the remote system.

For successful completion of the command:

- The transactions must be stopped.
- When assigning a remote transaction to local, a local APPLCTN macro (no SYSID parameter) for the corresponding program (PSB) must have been included in the system definition. There must be a local as well as a remote PSB of the same name defined during system definition.

`/MSASSIGN TRANSACTION` cannot be used for the CPI Communications driven transaction program.

Examples

Example 1 for /MSASSIGN Command

Entry ET:

```
/MSASSIGN LINK 2 TO MSPLINK BBB
```

Response ET:

```
DFS058I  MSASSIGN COMMAND COMPLETED
```

Explanation: The input system now has a logical connection to physical link BBB for all the SYSIDs assigned to link 2.

Example 2 for /MSASSIGN Command

Entry ET:

```
/MSASSIGN MSNAME BOSTON TO LINK 3
```

Response ET:

```
DFS058I  MSASSIGN COMMAND COMPLETED
```

Explanation: The MSNAME BOSTON and its associated SYSID (19) have been reassigned to link 3. Link 3 can be used to associate all its SYSIDs with a main storage-to-main storage connection as opposed to a channel-to-channel or a bisynchronous connection.

Example 3 for /MSASSIGN Command

Entry ET:

```
/MSASSIGN SYSID 19 TO LINK 2
```

Response ET:

```
DFS058I  MSASSIGN COMMAND COMPLETED
```

Explanation: SYSID 19 might have been originally assigned to link 1 in the remote system and known by the MSNAME of CHICAGO. The example above reassigns SYSID 19 (CHICAGO) to link 2. This assignment can be made when both link 1 and link 2 are connected to the same two IMS systems and link 1 is inoperable.

Example 4 for /MSASSIGN Command

Entry ET:

```
/MSASSIGN TRANSACTION XYZ TO MSNAME LNKY
```

Response ET:

```
DFS058I  MSASSIGN COMMAND COMPLETED
```

Explanation: Local and remote SYSIDs of transaction XYZ are assigned to the link associated with MSNAME LNKY.

Chapter 37. /MSVERIFY

Format

Environments and Keywords

Table 78 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 78. Valid Environments for the /MSVERIFY Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/MSVERIFY	X		X
MSNAME	X		X
SYSID	X		X

Usage

/MSVERIFY verifies local transactions and local LTERMs for corresponding remote transactions and remote LTERMs. It also verifies the consistency of the attributes of corresponding transactions. For instance, Fast Path exclusive transactions cause an error message to be returned because the transactions can only be processed as local transactions. Defined paths that are usable between the two systems can also be verified with this command.

The IMS multiple systems verification utility is run offline following system definition. /MSVERIFY is used during online execution when verification of two systems is desired, such as after an /MSASSIGN command has been entered.

MSNAME
Specifies the logical link path in a multiple systems configuration.

SYSID
Specifies the remote system identification of a system in a multiple systems configuration.

Restrictions: The /MSVERIFY command has the following restrictions:

- Only one remote system can be specified for each /MSVERIFY command.
- The /MSVERIFY command does not apply to ISC or directed routing.
- If an IMS Version 4 system is connected to a shared-queues environment, the /MSVERIFY command will only work if it is entered from the IMS system that is directly connected to the IMS Version 4 system.

Examples

Example 1 for /MSVERIFY Command

Entry ET:

```
/MSVERIFY MSNAME ABC
```

© Copyright IBM Corp. 1974, 2003

Response ET:

```
DFS2234I MSVERIFY COMMAND IN PROGRESS FOR LOCAL
SYSIDS 001,002,003,004 dddddd
```

Explanation: Assume SYSID numbers 001 through 004 are defined as local in the input system. dddddd is the time stamp.

Response ET:

```
DFS2243 TA11A IS NOT DEFINED AS TRANSACTION
IN BOTH SYSTEMS.
```

Explanation: TA11A is defined as a remote transaction in one system but is not defined as a transaction in the partner system.

Response ET:

```
DFS2237I MSVERIFY COMPLETED FOR LOCAL SYSID 001
dddddd
```

Explanation: Verification of local SYSID 001 is completed. The response message is repeated for local SYSIDs 002, 003, and 004 if all verify. dddddd is the time stamp.

Response ET:

```
DFS2235I  SYSID 004 IS DEFINED AS LOCAL IN BOTH
SYSTEMS.  dddddd
```

Explanation: SYSID 004 has no corresponding remote SYSID. dddddd is the time stamp.

Remote SYSIDs are processed next.

Response ET:

```
DFS2236I MSVERIFY COMMAND IN PROGRESS FOR REMOTE
SYSIDS 005,006,007,008 dddddd
```

Response ET:

```
DFS2237I MSVERIFY COMPLETED FOR SYSID 005
dddddd
```

Explanation: The verification of one remote SYSID is completed. The response above is repeated for remote SYSIDs 006, 007, and 008 if all verify. dddddd is the time stamp.

Example 2 for /MSVERIFY Command

Entry ET:

```
/MSVERIFY SYSID 8
```

Explanation: The input system sends a list of its local SYSIDs to the remote system and then displays the following message on the input terminal.

Response ET:

```
DFS2234I MSVERIFY COMMAND IN PROGRESS FOR LOCAL SYSIDS
001, 002, 003, 004 dddddd
```

Explanation: Assume SYSID numbers 001 through 004 are defined as local in the input system.

As the input system completes processing of the information returned for each of the input system's local SYSIDs, the following message is displayed on the input terminal:

Response RT:

```
DFS2237I MSVERIFY COMPLETED FOR SYSID 001
dddddd
```

Explanation: The verification of local SYSID 001 is completed. The response message is repeated for local SYSIDs 002, 003, 004 if all verify. Verification is complete only if a DFS2237I message is received for every SYSID identified in the DFS2234I message. Definition and assignment errors can cause conditions that prevent these messages from being returned. Stopped or unassigned links can also prevent the return of the message. dddddd is the time stamp.

Remote SYSIDs are processed next. The input system sends all its pertinent information related to the remote system's local SYSIDs and displays the following message:

Response ET:

```
DFS2236I MSVERIFY COMMAND IN PROGRESS FOR REMOTE
SYSIDs 005, 006, 007, and 008 dddddd
```

As the remote system completes processing of the information it received for each of its local SYSIDs, the following message is displayed on the input terminal. dddddd is the time stamp.

Response ET:

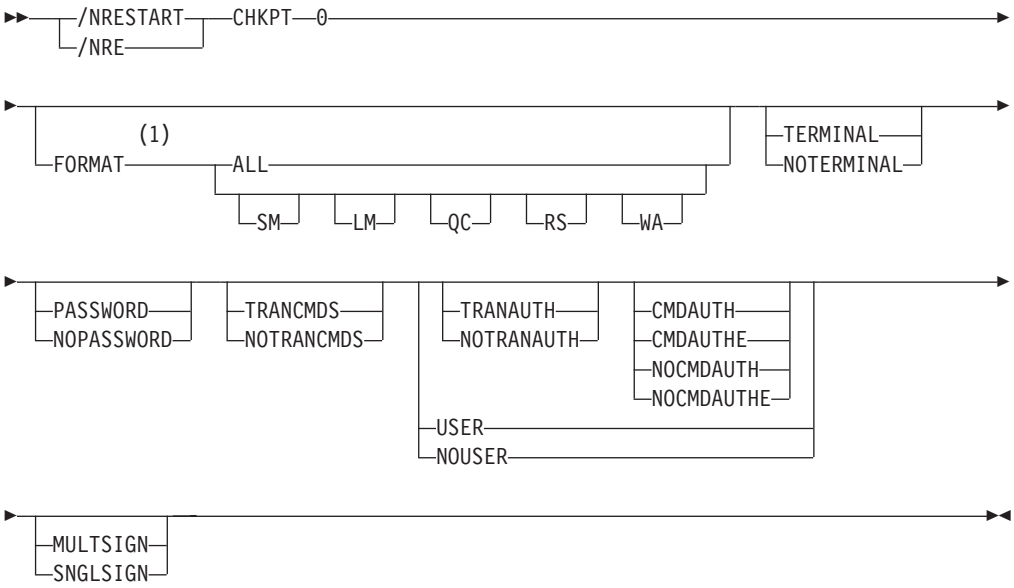
```
DFS2237I MSVERIFY COMPLETED FOR SYSID 005
dddddd
```

Explanation: The verification of one remote SYSID is complete. The last response message is repeated for remote SYSIDs 006, 007, 008 if all verify. dddddd is the time stamp.

Chapter 38. /NRESTART

Format

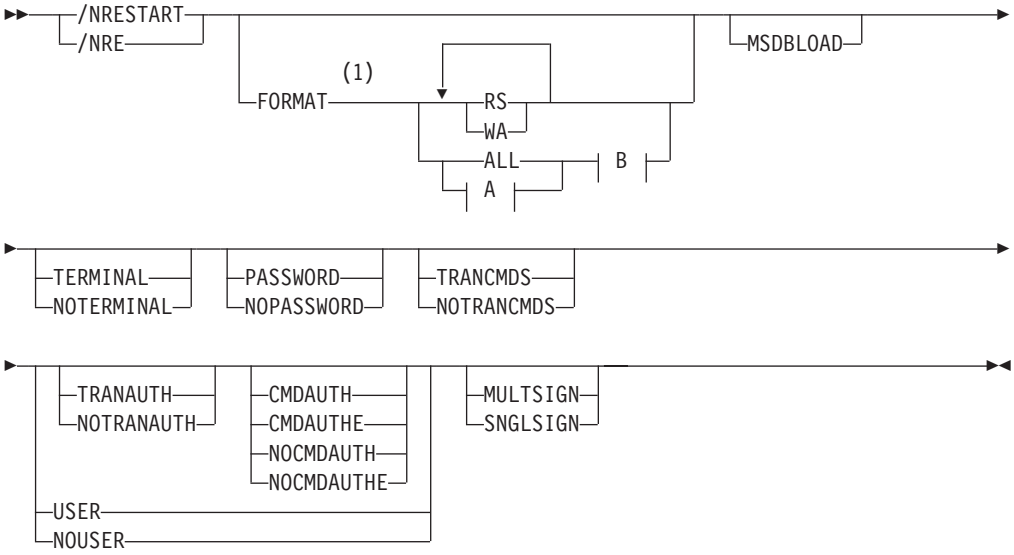
Cold Start With No Previous Shutdown



Notes:

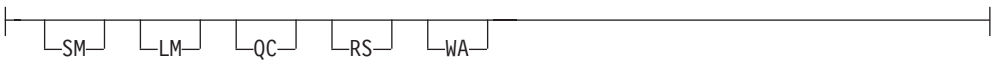
1 The FORMAT keyword must be followed by at least one of the SM, LM, QC, RS, WA, or ALL parameters.

Warm Start After a /CHECKPOINT FREEZE Command

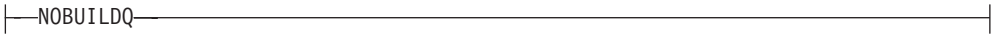


© Copyright IBM Corp. 1974, 2003

A:



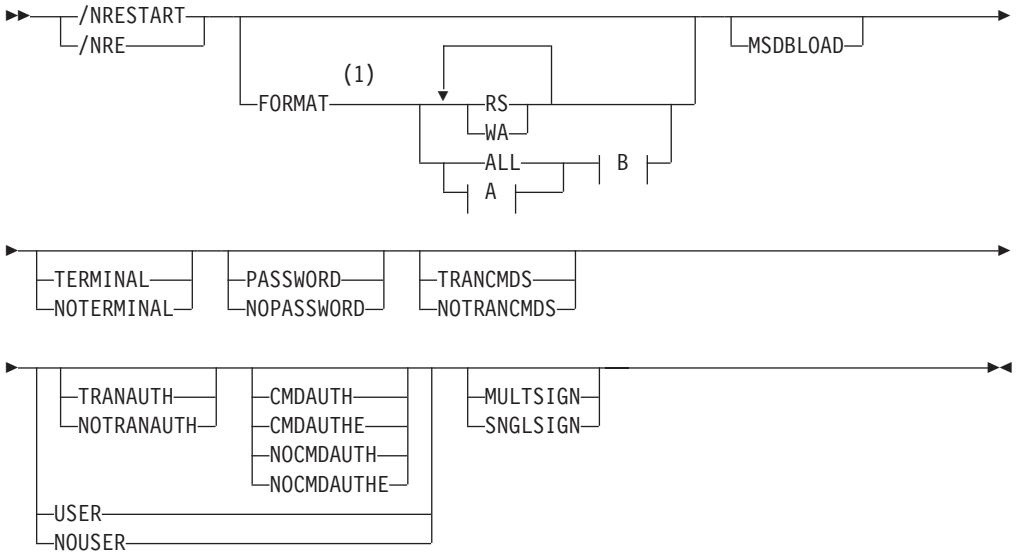
B:



Notes:

1 The FORMAT keyword must be followed by at least one of the SM, LM, QC, RS, WA, or ALL parameters.

Warm Start After a /CHECKPOINT PURGE or /CHECKPOINT DUMPQ Command



A:

B:

Notes:

1 The FORMAT keyword must be followed by at least one of the SM, LM, QC, RS, WA, or ALL parameters.

Environments and Keywords

Table 79 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 79. Valid Environments for the /NRESTART Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/NRESTART	X	X	X
BUILDQ	X		X
CHKPT	X	X	X
CMDAUTH	X		X
CMDAUTHE	X		X
FORMAT	X	X	X
MSDBLOAD	X		
MULTSIGN	X		X
NOBUILDQ	X		X
NOCMDAUTH	X		X
NOCMDAUTHE	X		X
NOPASSWORD	X		X
NOTERMINAL	X		X
NOTRANAUTH	X		X
NOTRANCMDS	X		X
NOUSER	X		X
PASSWORD	X		X
SNGLSIGN	X		X
TERMINAL	X		X
TRANAUTH	X		X
TRANCMDS	X		X
USER	X		X

Usage

The /NRESTART command has three forms. The form depends on:

- Whether a cold start or warm start is required
- In the case of warm starts, whether the previous orderly shutdown was accomplished with:
  - /CHECKPOINT FREEZE
  - /CHECKPOINT PURGE or DUMPQ

/NRESTART is used to cold start IMS or warm start IMS following an orderly termination accomplished with a /CHECKPOINT shutdown command.

Attention: A cold start performed after a processing failure could cause processing against uncommitted data. To ensure data integrity, be sure necessary backout or recovery operations have been performed before restarting.

When IMS initializes, the system parameters used for this initialization can come from the IMS system generation, from a PROCLIB member, or from EXEC statements that can override both the defaults and the PROCLIB members. Therefore, message DFS1929I is displayed showing the system parameters used for this particular initialization. The system parameters are also written to the job log.

For an IMS cold start, the base security definition is created from the IMS system definition and EXEC parameter specifications. For an /NRESTART warm restart, the base security definition is created from the IMS checkpoint data.

To override the base security definitions on a cold start, the security keywords of the /NRESTART command must be used.

The SGN=, TRN=, and RCF= startup parameters can be overridden by the /NRESTART command using the security keywords shown in Table 80 A brief description of the keywords is also included.

Table 80. Security Keywords and Their Startup Parameter Equivalents

Keyword	Description	Startup Parameter
CMDAUTH	RACF command authorization on static and ETO terminals only.	RCF=S
CMDAUTHE	RACF command authorization on ETO terminals only.	RCF=C
MULTSIGN	Permits multiple signons for each user ID.	SGN=M
NOCMDAUTH	Resets the command authorization on static and ETO terminals.	Not RCF=S
NOCMDAUTHE	Resets the command authorization on ETO terminals only.	Not RCF=C
NOTRANAUTH	Resets the transaction authorization.	Not TRN=F or Y
NOUSER	Resets user identification verification, transaction authorization, and command authorization.	Not SGN=F or Y (G or Z becomes M) Not TRN=F or Y Not RCF=C or S
SNGLSIGN	Permits a single signon for each user ID.	SGN=F and Y Not SGN=M (G or Z becomes F or Y)
TRANAUTH	Transaction authorization.	TRN=F or Y
USER	Sets user identification verification.	SGN=Y

BUILDQ or **NOBUILDQ**

BUILDQ requests that the message queues dumped on the log be loaded into the message queue data sets. BUILDQ is optional for a warm start after a /CHECKPOINT PURGE or /CHECKPOINT DUMPQ.

The BUILDQ keyword must be included if the message queues are to be restored after being formatted. If the BUILDQ keyword is specified, the log from the last /CHECKPOINT DUMPQ or /CHECKPOINT PURGE is required, slowing down the restart process.

If /NRESTART FORMAT is specified without the BUILDQ keyword, the NOBUILDQ keyword must be specified. specifying NOBUILDQ reformats the queues in question and all messages are lost.

If an /NRESTART BUILDQ command fails, and then the /ERESTART CHECKPOINT 0, /ERESTART COLDCOMM, or /ERESTART COLDSYS command is performed, the

messages are lost. program product (5655-038) can be used to recover the local message queues. For more information about MRQ, refer to the *IMS/ESA Message Requeuer Program Description/Operations Manual*.

In a shared-queues environment, the BUILDQ keyword is ignored because the message queue data sets are not used.

CHECKPOINT
Identifies the shutdown/restart sequence. `CHECKPOINT 0` must be specified for a cold start.

CMDAUTH
Specifies that both signon (user identification verification) and command authorization for static and ETO terminals are in effect at the end of the emergency restart. (Command authorization is same as specifying RCF=S on the startup parameter.)

To specify CMDAUTH, either:

```
SECURITY macro TYPE=RACFTERM|RACFCOM|SIGNEXIT|TRANEXIT
```

must be specified in the IMS system definition, or

```
EXEC parameters RCF=A|Y|T|C|S
```

must be specified on an EXEC parameter.

CMDAUTHE
Specifies that command authorization for ETO terminals (same as RCF=S on the startup parameter) is in effect at the end of the emergency restart. CMDAUTHE also resets command authorization for static terminals, if it was set.

To specify CMDAUTHE, either:

```
SECURITY macro TYPE=RACFTERM|RACFCOM|SIGNEXIT|TRANEXIT
```

must be specified in the IMS system definition, or

```
EXEC parameters RCF=A|Y|T|C|S
```

must be specified on an EXEC parameter.

FORMAT
Specifies which queues or data sets should be formatted as part of the restart process when:

- A message queue or data set I/O error occurs.
- The size of a message queue or data set is to be changed.
- A message queue or data set is to be reallocated.

Specify one or more of the following or ALL:

SM Short-message queue

LM Long-message queue

QC Control record data set

RS Restart data set

WA Write-ahead data set

ALL All message queues (SM and LM) and data sets (QC, WA, and RS)

When FORMAT ALL is specified, do not also specify SM, LM, QC, WA, or RS. FORMAT ALL is only required at IMS initialization (first time use of the system).

You can specify any combination of SM, LM, QC, WA, and RS; for example, FORMAT LM RS.

In a shared-queues environment, the LM, SM, and QC parameters are ignored because the message queue data sets are not used. If you specify ALL, IMS does not attempt to format the message queue data sets.

Table 81 shows the environments in which the parameters are valid.

Table 81. /NRESTART FORMAT Command Parameter Environments

Parameter	DB/DC	DBCTL	DCCTL
SM	X		X
LM	X		X
QC	X		X
RS	X	X	X
WA	X	X	X
ALL	X	X[1]	X

Note:

1. Supports only RS and WA parameters.

MSDBLOAD
Requests that the MSDBs be loaded from the z/OS sequential data set MSDBINIT instead of the MSDB checkpoint data set. Use the MSDBLOAD keyword only when an MSDB initial load is required; otherwise, omit it from the /NRESTART command. After you modify an MSDB DBD, you must specify MSDBLOAD on the next warm start of IMS in order for the changes to be effective.

MSDBLOAD is not required:

- For warm starts when the MSDB checkpoint data set is used.
- For a cold start because the MSDBs are loaded from the z/OS sequential data set MSDBINIT and the MSDB checkpoint data sets are formatted.

MULTSIGN
Permits multiple signons for each user ID.

In an IMSplex with Resource Manager and a resource structure, if MULTSIGN conflicts with the single user signon definition for the IMSplex, a warning message will be issued.

NOCMDAUTH
Resets command authorization on static and ETO terminals.

NOCMDAUTHE
Resets command authorization for static and ETO terminals. The command authorization is reset for static terminals because the command authorization for static terminals cannot exist without the command authorization for ETO terminals.

NOTERMINAL
Specifies that the terminal security specifications established by the Security Maintenance utility are not in effect at completion of this emergency restart.

IMS system definition might have precluded the authority of the master terminal operator to negate terminal security, in which case, if NOTERMINAL is specified, an error message is received.

NOTRANAUTH
Turns off transaction authorization. NOTRANAUTH is not the opposite of TRANAUTH. TRANAUTH sets transaction authorization and also turns on signon (user identification verification).

If you specify NOTRANAUTH, it will be rejected with an error message if either:

- SECLEVEL=FORCTRAN was specified on the system definition SECURITY macro.
- TRN=F was specified as a JCL EXEC parameter.

NOUSER
Specifies that none of the following is in effect at the end of the emergency restart:

- Transaction authorization
- User identification verification
- Command authorization

You can use `/NRESTART NOUSER` for a warm start, but if transaction authorization or command authorization are set from the checkpoint data, NOUSER is ignored, and signon (user identification verification) is set on.

PASSWORD or **NOPASSWORD**
Specifies whether (PASSWORD) or not (NOPASSWORD) the password security specifications established by the Security Maintenance utility will be in effect at completion of this normal restart.

IMS system definition might preclude the authority of the master terminal operator to negate password security. If this is the case, and you specify NOPASSWORD, you receive an error message.

SNGLSIGN
Permits a single signon for each user ID.

In an IMSplex with Resource Manager and a resource structure, if SNGLSIGN conflicts with the single user signon definition for the IMSplex, a warning message will be issued.

TERMINAL
Specifies that the terminal security specifications established by the Security Maintenance utility are in effect at completion of this emergency restart. if RCF=S is specified. SMU is not used for command terminal security if RACF is used.

TRANAUTH
Specifies both transaction authorization and user identification verification, with or without RACF.

To specify TRANAUTH either:

```
SECURITY macro TYPE=RACFTERM|RACFCOM|SIGNEXIT|TRANEXIT
```

must be specified in the IMS system definition, or

```
EXEC parameters RCF=A|Y|T|C|S
```

must be specified on an EXEC parameter.

TRANCMDS or **NOTRANCMDS**
Specifies whether (TRANCMDS) or not (NOTRANCMDS) the transaction command security established by the Security Maintenance utility is in effect at completion of this normal restart. NOTRANCMDS specifies that transaction-entered commands cannot be entered.

USER
Specifies user identification verification. User identification verification means that signon is required by the static terminals that are defined to SMU for signon verification. This keyword has no effect on ETO terminals because they are always required to sign on. User identification verification can be forced on by the TRANAUTH or CMDAUTH keyword

To specify USER either:

```
SECURITY macro TYPE=RACFTERM|RACFCOM|SIGNEXIT|TRANEXIT
```

must be specified in the IMS system definition, or

```
EXEC parameters RCF=A|Y|T|C|S
```

must be specified on an EXEC parameter.

Examples

Example 1 for /NRESTART Command

This is an example of a cold start with new message queue data sets.

Entry ET:

```
/NRESTART CHECKPOINT 0 FORMAT ALL
```

Response ET:

```
DFS058I  (time stamp) NRESTART COMMAND IN PROGRESS
DFS994I *CHKPT 82274/114447**SIMPLE*
```

Explanation: IMS is started at 114447 (time) on 82274 (Julian date). A simple checkpoint is written on the system log. All message queue data sets are formatted. 82274/114447 is the checkpoint number.

Example 2 for /NRESTART Command

This is an example of a warm start from a FREEZE checkpoint.

Entry ET:

```
/NRESTART
```

Response ET:

```
DFS058I  (time stamp) NRESTART COMMAND IN PROGRESS
DFS680I USING CHKPT 82273/180000
DFS994I *CHKPT 82274/082217**SIMPLE*
```

Explanation: The restart is being performed from checkpoint 82273/180000, which was written at the most recent IMS shutdown. IMS is restarted at 082217 (time) on 82274 (Julian date). A simple checkpoint is written on this system log. 82274/082217 is the checkpoint number.

Example 3 for /NRESTART Command

This is an example of a warm start to format WADS.

Entry ET:

```
/NRESTART FORMAT WA
```

Response ET:

```
DFS058I  (time stamp) NRESTART COMMAND IN PROGRESS
DFS680I USING CHKPT 82119/230000
DFS994I *CHKPT 82120/101318**SIMPLE*
```

Explanation: The restart is being performed from checkpoint 82119/230000, which was written at the most recent IMS shutdown. IMS is restarted at 101318 (time) on 82120 (Julian date). A simple checkpoint is written on the system log. 82120/101318 is the checkpoint number.

Example 4 for /NRESTART Command

This is an example of a warm start from a PURGE or DUMPQ checkpoint.

Entry ET:

```
/NRESTART BUILDQ
```

Response ET:

```
DFS058I  (time stamp) NRESTART COMMAND IN PROGRESS
DFS680I USING CHKPT 82080/214240
DFS994I *CHKPT 82081/060000**SIMPLE*
```

Explanation: IMS is restarted at 060000 (time) on 82081 (Julian date) from checkpoint 82080/214240, which was written at the most recent IMS shutdown. 82081/060000 is the checkpoint number.

Example 5 for /NRESTART Command

This is an example of a warm start from a PURGE or DUMPQ checkpoint. The large and small message queue data sets have been reallocated.

Entry ET:

```
/NRESTART BUILDQ FORMAT SM LM
```

Response ET:

```
DFS058I  (time stamp) NRESTART COMMAND IN PROGRESS
DFS680I USING CHKPT 82170/085236
DFS994I *CHKPT 82170/085820**SIMPLE*
```

Explanation: IMS is restarted at 085820 (time) on 82170 (Julian date) from checkpoint 82170/085236, which was written at the most recent IMS shutdown. The large and small message queue data sets are reformatted. 82170/085820 is the checkpoint number.

Example 6 for /NRESTART Command

This is an example of a warm start from a PURGE or DUMPQ checkpoint. An initial set of MSDBs is needed.

Entry ET:

```
/NRESTART BUILDQ MSDBLOAD
```

Response ET:

```
DFS058I  (time stamp) NRESTART COMMAND IN PROGRESS
DFS680I USING CHKPT 82068/180000
DFS2554 MSDB MSDBHJ01 LOADED
DFS2554 MSDB MSDBHJ02 LOADED
DFS2554 MSDB MSDBAK01 LOADED
DFS2554 MSDB MSDBAK02 LOADED
DFS2554 MSDB MSDBPS01 LOADED
DFS994I *CHKPT 82069/080000**SIMPLE*
```

Explanation: IMS is restarted at 080000 (time) on 82069 (Julian date) from checkpoint 82068/180000, which was written at the most recent IMS shutdown. A simple checkpoint is written on the system log. 82069/080000 is the checkpoint number. An initial set of MSDBs is loaded from the z/OS sequential data set MSDBINIT.

Example 7 for /NRESTART Command

This is an example of a warm start from a PURGE or DUMPQ checkpoint with a request for transaction command security.

Entry ET:

```
/NRESTART TRANCMDS
```

Response ET:

```
DFS058I  (time stamp) NRESTART COMMAND IN PROGRESS
DFS680I USING CHKPT 82080/214240
DFS994I *CHKPT 82274/114447**SIMPLE*
```

Explanation: The master terminal operator is warm starting IMS and requesting that transaction command security be in effect at the completion of the normal restart.

Example 8 for /NRESTART Command

This is an example of a warm start from a PURGE or DUMPQ checkpoint with a request to negate transaction command security.

Entry ET:

```
/NRESTART NOTRANCMDS
```

Response ET:

```
DFS2181I  CANNOT OVERRIDE FORCED COMMAND SECURITY
```

Explanation: The master terminal operator is warm starting IMS with a request to negate transaction command security. IMS system definition precludes the authority of the master terminal operator to enter the NOTRANCMDS keyword. IMS returns the DFS2181 message.

Example 9 for /NRESTART Command

This is an example of a warm start from a PURGE or DUMPQ checkpoint. The data sets are formatted without rebuilding the message queues.

Entry ET:

```
/NRESTART FORMAT SM LM NOBUILDQ
```

Response ET:

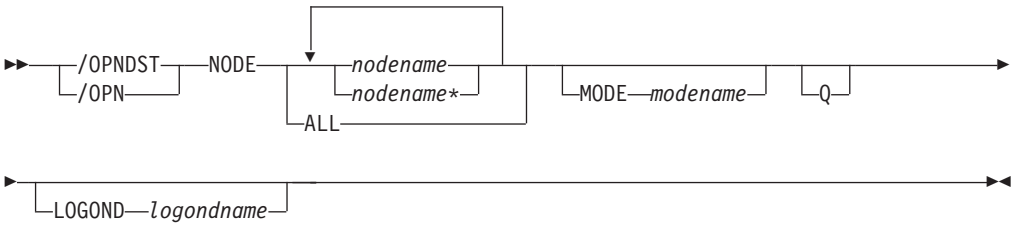
```
DFS058 NRESTART COMMAND IN PROGRESS
```

Explanation: The master terminal operator is warm starting IMS with a request to reformat the data sets without rebuilding the message queues. IMS comes up, but all messages are lost.

Chapter 39. /OPNDST

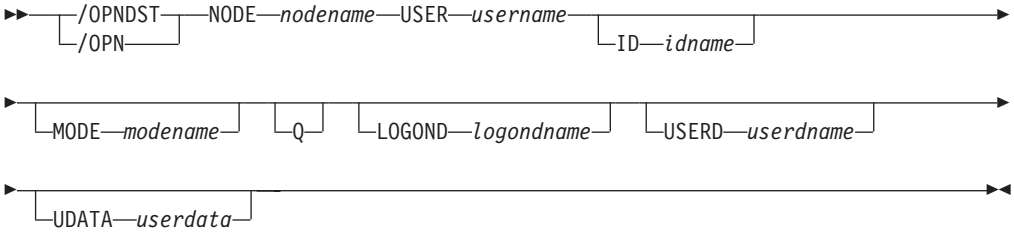
Format

The /OPNDST Command Without the USER Keyword. Use this form of the command for all static and ETO terminals except ETO SLU P and Finance terminals, ETO output-only devices, and all ISC parallel sessions.

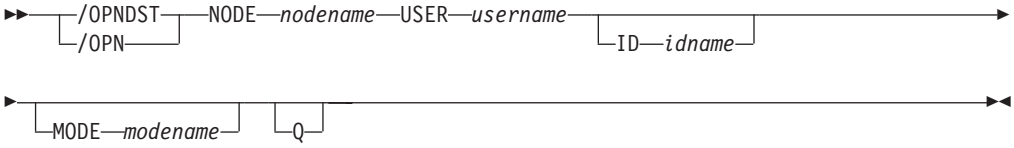


The /OPNDST Command With the USER Keyword for Non-ISC. Use this form of the command for:

- ETO SLU P and Finance terminals
- ETO output-only devices, for example, 3284, 3286, and SLU P1 with a single component of PRINTER1



The /OPNDST Command With the USER Keyword for ISC. Use this form of the command for all ISC parallel sessions, both static and ETO.



Environments and Keywords

Table 82 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 82. Valid Environments for the /OPNDST Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/OPNDST	X		X
ID	X		X
LOGOND	X		X

© Copyright IBM Corp. 1974, 2003

Table 82. Valid Environments for the /OPNDST Command and Keywords (continued)

Command / Keywords	DB/DC	DBCTL	DCCTL
MODE	X		X
NODE	X		X
Q	X		X
UDATA	X		X
USER	X		X
USERD	X		X

Usage

/OPNDST is a multisegment command that causes IMS to initiate a session with a VTAM terminal and, if the USER keyword is specified, the user is signed on automatically to the VTAM terminal after successful session initiation. All forms of logging-on a remote VTAM terminal to IMS, including the use of /OPNDST, do not work until the /START DC command has been entered and accepted by IMS.

All /OPNDST formats require an EOM indication to denote end-of-message. An EOS indication must be included for all segments that precede the last segment. See “Multisegment Command Input” on page 8 for more detail on using EOM and EOS.

You can issue /OPNDST on the XRF alternate to restart a failed backup session for a class 1 ETO terminal. To do this, the node and the user structure must still exist and be coupled together, and an active session must exist on the active system. If the node and the user are not coupled, or an active session does not exist on the active system, the command is rejected.

ID Is applicable only if the USER keyword is specified and the node is a parallel session ISC node. ID identifies the other system half-session qualifier. ID must not be specified for a single session ISC node. ID idname must be specified to open ISC nodes defined with users. The ID idname is passed to the other half-session with the session initiation request. If the other system is another IMS system, idname is the name of an ISC user in that system.

LOGOND
Indicates the logon descriptor used for session establishment. LOGOND is supported only for dynamic non-ISC nodes. The logon descriptor can also be provided through the logon exit.

MODE
Identifies the LOGON MODE table entry that VTAM must use and can determine operating characteristics for certain VTAM terminals. If a list of node names is given, the MODE keyword is applied to each of them. The command checks whether each node has been defined to accept IMS initiated connections. If the node was defined as NOPNDST, the /OPNDST command accepts all the nodes except the node defined as NOPNDST.

NODE
Specifies the VTAM node with which IMS will initiate a session.

/OPNDST NODE ALL opens sessions for all static terminals except ISC parallel sessions. The command has considerable concurrent activity, both for IMS and for VTAM. Ensure that the system has sufficient pool sizes, buffer sizes, and number of concurrent IMS tasks defined.

If the USER keyword is not specified in the command, the NODE parameter can be generic or ALL, or a range of static nodes, or there can be multiple NODE parameters. If a generic, ALL, or range of node names is specified, any nodes defined with users are ignored and flagged with an error message.

The /OPNDST NODE USER command signs on and logs on a user at the same time, except for ISC terminals. On ISC terminals, the session is allocated. Sample /OPNDST NODE USER commands are listed below:

- To sign on and log on to a static non-ISC terminal:

  ```
  /OPNDST NODE nodename USER username UDATA userdata
  ```

  This command marks a statically defined terminal as signed on by the user.
- To sign on and log on to an ETO non-ISC terminal:

  ```
  /OPNDST NODE nodename USER username MODE modename
  LOGOND logondname USERD userdname UDATA userdata
  ```

  This command creates the terminal and user structures, and allocates the newly created user structure to the terminal structure created to indicate signed on status.
- To sign on and log on to a static ISC terminal:

  ```
  /OPNDST NODE nodename USER username ID idname
  ```

  This command finds the subpool structure and allocates the subpool (user) structure the statically defined static ISC terminal. user issues the /SIGN command.
- To sign on and log on to an ETO ISC terminal:

  ```
  /OPNDST NODE nodename USER username ID idname MODE modename
  LOGOND logondname USERD userdname UDATA userdata
  ```

  This command creates the terminal and subpool (user) structure and allocates the newly created user structure to the terminal structure created. defined by the SUBPOOL name in RACF until the user issues a /SIGN command.

To restart failing ETO sessions (for example, ETO only or printer sessions), use the /OPNDST NODE USER command.

In an IMSplex, /OPNDST NODE specifies a VTAM node with which IMS initiates a session. Specify ROUTE(imsid), if you want to log the node onto a particular IMS. If ROUTE(imsid) is not specified, and /OPNDST is routed to all the IMS systems, IMS will process the command only on the IMS system designated as the command master. If the ROUTE keyword specifies multiple IMS systems so that the /OPNDST is routed to more than one IMS systems, IMS will process the command only on the IMS system designated as the command master. On the other IMSs, the /OPNDST command will be rejected.

Q Causes IMS to request VTAM to queue SIMLOGON requests for VTAM/SNA-supported terminals.

The /OPNDST NODE Q command also allows IMS to request another subsystem to share a node (usually printers) with IMS. If the other subsystem is using the printer, VTAM queues the SIMLOGON request for IMS for the printer, schedules the owning subsystem's RELREQ VTAM exit, and acquires the printer for IMS after the current owning system releases the printer. Multiple requests for the same printer are queued by VTAM for processing.

The /DISPLAY NODE command is used to determine whether IMS has acquired the printer.

UDATA
Indicates the user data used with the signon.

The UDATA keyword is valid only if the USER keyword and parameter are also specified. The UDATA keyword is valid for static and dynamic users. It is not valid for ISC nodes. The user data can be up to 256 bytes long.

USER
Identifies the logical terminal user to be allocated to the half-session to be created for ISC node nodename. For dynamic non-ISC users, it specifies the user ID to be signed on to the dynamic node nodename.

The USER keyword applies to ISC sessions when allocating a user to an ISC node, to dynamic users when signing a dynamic user on to a dynamic node, and to static user IDs when signing a user on to a static node.

USER username must be specified to open parallel session ISC nodes with users. It must not be specified for a single session ISC node.

USERD
Specifies the user descriptor to be used with the signon. It is valid only if the USER keyword and parameter are specified. USERD is only supported for dynamic users and is only valid for dynamic non-ISC nodes. The user descriptor can also be provided through the logon or signon exits.

Examples

Example 1 for /OPNDST Command

Entry ET:

```
/DIS NODE L3270*
```

Response ET:

```
NODE-USR TYPE   CID       RECD ENQCT DEQCT   QCT  SENT
L3270A   3277   00000000     0     0     0     0     0 IDLE C1INOP STATIC
L3270B   3277   00000000     0     0     0     0     0 IDLE C1INOP STATIC
L3270C   3277   08000002    44    45    45     0    80 CON STATIC
L3270D   3277   00000000     2     0     0     0     7 IDLE STATIC
*94307/145048*
```

Entry ET:

```
/OPNDST NODE L3270*
```

Response ET:

```
DFS058I  OPNDST COMMAND COMPLETED
```

Entry ET:

```
/DIS NODE L3270*
```

Response ET:

```
NODE-USR TYPE   CID       RECD ENQCT DEQCT   QCT  SENT
L3270A   3277   06000004     0     0     0     0     1 IDLE CON STATIC
L3270B   3277   04000005     0     0     0     0     1 IDLE CON STATIC
L3270C   3277   08000002    46    46    46     0    82 CON STATIC
L3270D   3277   04000006     2     0     0     0     8 IDLE CON STATIC
```

```
  *94307/145750*

Explanation:  The nodes L3270A through L3270D are logged on to IMS.
```

Example 2 for /OPNDST Command

Entry ET:

```
   /OPNDST NODE DT327002 USER IMSUS01 MODE LU032NT4 USERD DFSUSER
   UDATA= IMSPW01.
```

Response ET:

```
  DFS058I 11:07:48 OPNDST COMMAND COMPLETED
```

Explanation: A session with dynamic node DT327002 is established by using mode table LU032NT4. User IMS01 is signed on to the node using user descriptor DFSUSER, username IMSUS01, and password IMSPW01.

Example 3 for /OPNDST Command

Entry ET:

```
  /OPNDST NODE DTSLU201 USER IMSUS01 MODE SLU2MOD1 USERD DFSUSER
  LOGOND DFSSLU2 UDATA=IMSPW01.
```

Response ET:

```
  DFS058I 11:07:48 OPNDST COMMAND COMPLETED
```

Explanation: DTSLU201 is logged on. A session with dynamic node DTSLU201 is established by using logon descriptor DFSSLU2 (type SLU2), and mode table SLU2MOD1. Dynamic user IMSUS01 is signed on to the node using user descriptor DFSUSER and password IMSPW01.

Example 4 for /OPNDST Command

Entry ET:

```
  /OPNDST NODE WEST-EAST
```

Response ET:

```
  DFS058I  OPNDST COMMAND COMPLETED
```

Response RT:

```
  DFS3650 TERMINAL CONNECTED TO IMS XXXXXXXX
```

Explanation: The nodes, WEST through EAST, are logged on to IMS.

Example 5 for /OPNDST Command

The following set of commands illustrate the use of the MODE keyword on the /OPNDST command.

Entry ET:

```
  /DIS NODE LUTYPEP1 MODE
```

Response ET:

```
  NODE-USR TYPE   DEF MODETBL  ACT MODETBL
   LUTYPEP1 SLUP   DEFRESP
   *90179/100206*
```

Explanation: DEFRESP is the mode table name defined for node LUTYPEP1 at system definition. The session is not active so the ACT MODETBL field is blank.

Entry ET:

```
/OPN NODE LUTYPEP1.
```

Response ET:

```
DFS058I OPNDST COMMAND COMPLETED
```

Entry ET:

```
/DIS NODE LUTYPEP1 MODE
```

Response ET:

```
NODE-USR TYPE   DEF MODETBL  ACT MODETBL
 LUTYPEP1 SLUP   DEFRESP      DEFRESP
 *90179/100508*
```

Explanation: A mode table name was not specified with the /OPNDST command so the default value defined at system definition was used to initiate the session.

Entry ET:

```
/CLS NODE LUTYPEP1
```

Response ET:

```
DFS058I CLSDST COMMAND COMPLETED
```

Entry ET:

```
/DIS NODE LUTYPEP1 MODE
```

Response ET:

```
NODE-USR TYPE   DEF MODETBL  ACT MODETBL
 LUTYPEP1 SLUP   DEFRESP
 *90179/100630*
```

Explanation: Active mode table name displays as blank at normal session termination.

Entry ET:

```
/OPN NODE LUTYPEP1 MODE ALPHA.
```

Response ET:

```
DFS058I OPNDST COMMAND COMPLETED
```

Entry ET:

```
/DIS NODE LUTYPEP1 MODE
```

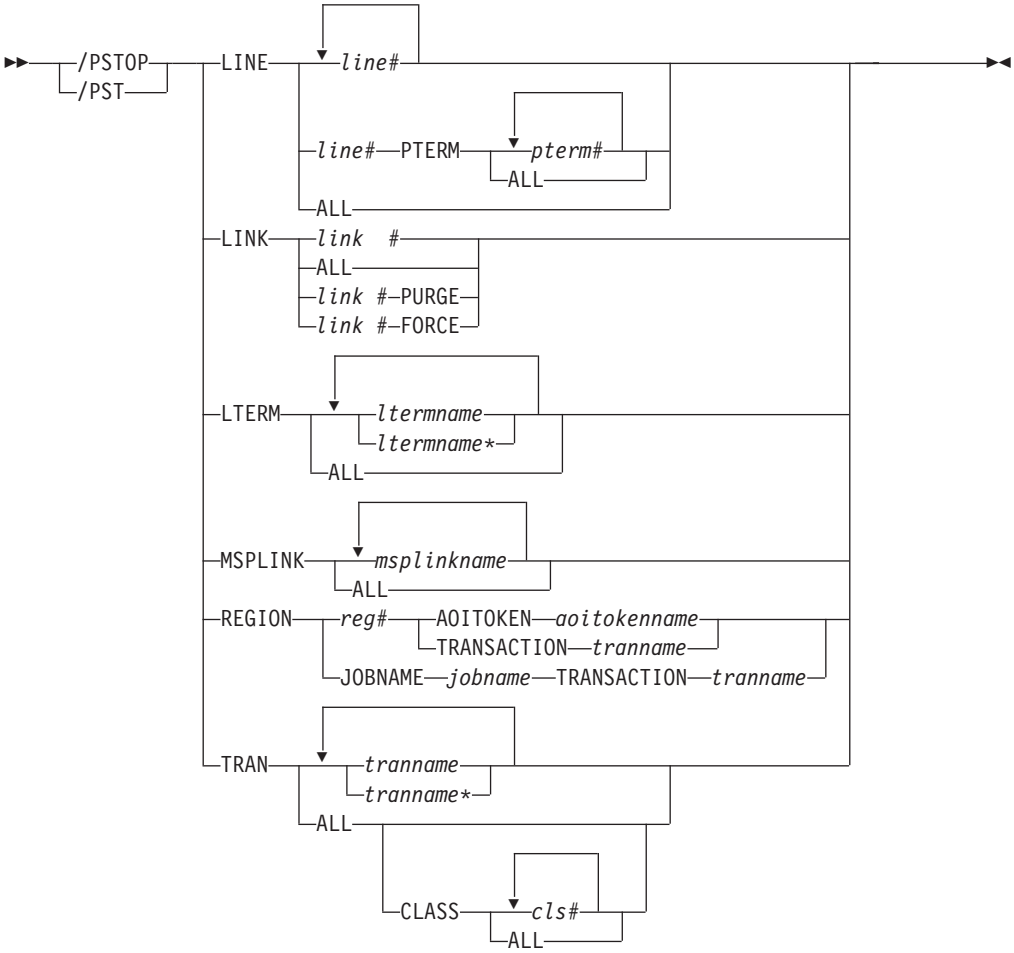
Response ET:

```
NODE-USR TYPE   DEF MODETBL  ACT MODETBL
 LUTYPEP1 SLUP  DEFRESP       ALPHA
 *90179/100805*
```

Explanation: The mode table name specified with the /OPNDST command (ALPHA) is used to initiate the session. The default value specified at system definition (DEFRESP) is overridden by the /OPNDST command.

Chapter 40. /PSTOP

Format



Environments and Keywords

Table 83 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 83. Valid Environments for the /PSTOP Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/PSTOP	X	X	X
AOITOKEN	X	X	X
CLASS	X		X
FORCE	X		X
JOBNAME	X	X	X
LINE	X		X
LINK	X		X
PURGE	X		X

© Copyright IBM Corp. 1974, 2003

Table 83. Valid Environments for the /PSTOP Command and Keywords (continued)

Command / Keywords	DB/DC	DBCTL	DCCTL
LTERM	X		X
MSPLINK	X		X
PTERM	X		X
REGION	X	X	X
TRAN	X		X

Usage

/PSTOP accomplishes the following:

- Stops the sending and receiving of messages to a particular communication line, terminal, or logical link
- Stops the scheduling of messages containing specific transaction codes
- Allows the queuing of output messages and input messages to continue
- Validity checks all parameters entered by the terminal operator

If an error is detected on parameters that are independent of one another, only the invalid parameters are indicated as being in error and the /PSTOP command processes the rest of the parameters.

/PSTOP can be used to reset conditions previously established with the /START, /RSTART, /PURGE, or /MONITOR command.

In a single IMS system, or in the local system in a multiple system configuration, IMS system messages such as broadcast text and terminal status messages (DFS059 TERMINAL STARTED) are not affected by /PSTOP. In a multiple system configuration, broadcast messages are queued but not sent across process stopped links. potential transactions.

LINE
Specifies the communication line that IMS will stop sending messages to and receiving messages from.

LINK
Specifies the link to be stopped; the partner link in another IMS system stops itself and notifies that system's master terminal operator.

/PSTOP LINK resets continuous mode for a BISYNC link. See the /RSTART LINK CONTINUOUS command in Chapter 51, "/RSTART," on page 573 for an explanation of continuous mode.

PURGE
PURGE can only be used for one logical link whose physical link is channel-to-channel. PURGE must be used when the partner link is in a system that has failed and the link will not go idle although it has been stopped.

FORCE
The FORCE keyword is for VTAM links and is intended for use when an MSC VTAM link will not clean up and idle during normal PSTOP processing, even though VTAM has terminated the session. It can be used, in conjunction with some VTAM commands to idle and clean up the VTAM link within IMS, so that an /RSTART LINK can be issued to restart the link.

The operation of the command first determines if the link has begun PSTOP processing. (If so, the link will display as PSTOPPED NOTIDLE). Next it will test if the hang condition is due to an outstanding VTAM request that has not completed. If so, it will issue an inquire request to VTAM to determine if the session is inactive. In that case, IMS will simulate completion of the VTAM request and allow the link to complete PSTOP processing.

To determine if the session is still active to VTAM, issue a DISPLAY NET,SESSIONS,LU1=applid1,LU2=applid2,SCOPE=ALL,LIST=ALL, and note the SID of the session if it is active.

If the session is not active, and it has a PSTOPPED NOTIDLE status to IMS, then /PSTOP LINK x FORCE can be issued. If the session is still active to VTAM, then VARY NET,TERM,SID=x,NOTIFY=YES,SCOPE=ALL,TYPE=FORCE can be issued to VTAM to terminate the session.

Under normal conditions, the VTAM VARY NET,TERM command should terminate the session in VTAM and cause IMS to PSTOP and IDLE the link associated with the VTAM session. If the VTAM VARY command does terminate the VTAM session but does not PSTOP and IDLE the link, then the IMS /PSTOP LINK FORCE command can be used to complete the PSTOP and cleanup processing within IMS.

After PSTOP processing is completed on one side of the line, the other side should be displayed. If the other side is not in a PSTOP IDLE state, the operation should be repeated for the other IMS. When both sides are in the PSTOP IDLE state, the /RSTART LINK command can be issued.

IMS replies with DFS058 /PSTOP LINK COMPLETED EXCEPT LINK x, if the /PSTOP LINK x FORCE command cannot be executed because:

- The session is still active to VTAM.
- Normal PSTOP processing has not begun (issue /PSTOP without the FORCE keyword in this case).
- PSTOP processing is not completing due to some reason other than an incomplete VTAM request.

LTERM

Specifies the logical terminal that is to be stopped from sending and receiving messages.

The /PSTOP LTERM command has no effect on an LTERM that is in QLOCK state, or is a remote logical terminal. The LTERM parameter can be generic, where the generic parameter specifies LTERMs that already exist.

The /PSTOP LTERM command is valid only for LTERMs that belong to nodes that are logged on.

MSPLINK

Applies to MSC VTAM links only; it stops logons to the physical link and enables the operator to reassign (/MSASSIGN) logical links to the physical link. The /MSASSIGN command does not affect links in sessions that have not been stopped by the /PSTOP command. When logical link assignments are complete, the /RSTART command should be issued to permit logons to the physical link.

REGION

If the TRANSACTION keyword is specified, the message region is not stopped. A QC status (no more messages) is returned to the application program currently active in the specified region. The scheduler will continue to schedule available transactions in the referenced region.

The /PSTOP REGION command is ignored unless both of the following occur:

- An active transaction type is specified.
- The referenced message region is processing transactions with the wait-for-input option, or the region is an MPP.

If the AOITOKEN keyword is specified, the AO application in wait AOI token state is posted and receives AIB return code X'00000004' and reason code X'0000004C'.

If JOBNAME keyword is specified, the job name for the dependent region must be 1-8 alphanumeric or national ($,#,@) characters. The first character of the job name must be either alphabetic or national.

TRANSACTION
Stops the scheduling of transactions; however, the transactions will continue to be processed until the limit count is reached. If the limit count is large, the processing interval will be long. The `/DISPLAY` command ascertains the status of the transaction; the `/ASSIGN` command alters the status of the transaction.

If a region is scheduled against a process stopped transaction and there are no more messages available for that transaction, the region does not wait for the next message (wait-for-input-mode). Instead, a QC status (no more messages) is returned to the application. If the region is scheduled and waiting for the next message when the command is entered, the region is notified and a QC status is returned to the application.

A batch message processing region (BMP) scheduled against wait-for-input (WFI) transactions returns a QC status code (no more messages) for `/PSTOP REGION`, `/DBD`, `/DBR`, or `/STA` commands only.

`/PSTOP` cannot stop the scheduling of Fast Path exclusive transactions but can be used to stop Fast Path potential transactions.

`/PSTOP TRANSACTION` cannot be used for Fast Path exclusive transactions or CPI Communications driven transaction programs.

The TRANSACTION parameter can be generic where the generic parameter specifies transactions that already exist.

Examples

Example 1 for /PSTOP Command

Entry ET:

```
/PSTOP LINE 4 PTERM 1
```

Response ET:

```
DFS058I  PSTOP COMMAND COMPLETED
```

Response RT:

```
DFS059I  TERMINAL PSTOPPED
```

Explanation: LINE 4 PTERM 1 is not sent application program or message switch output and is not allowed to send input. Output messages for the terminal continue to be queued.

Example 2 for /PSTOP Command

Entry ET:

```
/PSTOP LINE 4 6 200
```

Response ET:

```
DFS058I  PSTOP COMMAND COMPLETED EXCEPT LINE 200
```

Explanation: LINE 4 and LINE 6 are not allowed to send or receive messages. Message queuing continues. Line 200 is an invalid line number.

Example 3 for /PSTOP Command

Entry ET:

```
/PSTOP LINK 2 3 4
```

Response ET:

```
DFS058I  PSTOP COMMAND COMPLETED
```

Response ET:

```
DFS2169I  DISCONNECTION COMPLETED ON LINK 2
```

Explanation: Logical link 2 is disconnected. This message is received for each logical link that is disconnected.

Response Remote MT:

```
DFS2161I  LINK 2 STOPPED BY PARTNER
DFS2161I  LINK 3 STOPPED BY PARTNER
DFS2161I  LINK 4 STOPPED BY PARTNER
```

Explanation: Logical links 2, 3, and 4 stop processing messages. Output queuing continues.

Response ET:

```
DFS2169I  DISCONNECTION COMPLETED ON LINK 3
```

Explanation: Logical link 3 is disconnected. This message is also received when logical link 4 disconnects.

Example 4 for /PSTOP Command

Entry ET:

```
/PSTOP LINK ALL
```

Response ET:

```
DFS058I  PSTOP COMMAND COMPLETED
```

Response Remote MT:

```
A DFS2161 LINK n STOPPED BY PARTNER message
is received for each logical link that was
operational when /PSTOP was entered.
```

Explanation: Output to all logical links stops. Output queuing continues. Input is not allowed.

Response ET:

```
DFS2169I  DISCONNECTION COMPLETED ON LINK XXX
```

Explanation: As each logical link is disconnected, this message is received.

Example 5 for /PSTOP Command

Entry ET:

```
/PSTOP LINK 1 2 3 PURGE
```

Response ET:

```
DFS2272I PURGE KEYWORD INVALID, ONLY ONE
CTC LINK ALLOWED
```

Explanation: Only one link can be specified with the PURGE keyword.

Example 6 for /PSTOP Command

Entry ET:

```
/PSTOP LINK 2 PURGE
```

Response ET:

```
DFS2273I PURGE KEYWORD REJECTED, CURRENT STATUS
OF LINK IS NORMAL
```

Explanation: The partner system has not failed and the link appears to be working.

Example 7 for /PSTOP Command

Entry ET:

```
/PSTOP LTERM APPLE, TREE
```

Response ET:

```
DFS058I  PSTOP COMMAND COMPLETED
```

Response RT:

```
DFS059I  TERMINAL PSTOPPED
```

Explanation: The physical terminals associated with logical terminals APPLE and TREE are not sent output that is destined for logical terminals APPLE or TREE, or allowed to enter input. Output queuing continues.

Example 8 for /PSTOP Command

Entry ET:

```
/PSTOP MSPLINK ALL
```

Response ET:

```
DFS058I  COMMAND COMPLETED
```

Explanation: All the VTAM physical links are stopped from receiving logons. Any links in session are not affected.

Example 9 for /PSTOP Command

Entry ET:

```
/PSTOP TRANSACTION SEED
```

Response ET:

```
DFS058I  PSTOP COMMAND COMPLETED
```

Explanation: Transaction code SEED can no longer be scheduled. Queuing of the transaction continues.

Example 10 for /PSTOP Command

Entry ET:

```
/PSTOP TRANSACTION ALL CLASS 3
```

Response ET:

```
DFS058I  PSTOP COMMAND COMPLETED
```

Explanation: All transactions associated with class 3 can no longer be scheduled. Queuing of the transactions continues.

Example 11 for /PSTOP Command

Entry ET:

```
/PSTOP REGION 1 TRANSACTION XYZ
```

Response ET:

```
DFS058I  PSTOP COMMAND IN PROGRESS
```

Response ET:

```
DFS0569I PSTOP COMPLETE FOR REGION 1 TRAN XYZ
DFS0566I PSTOP NOT VALID FOR TRAN XYZ
```

Explanation: If the DFS0569I message prints, processing of the transaction type, xyz, is stopped in message region 1. If the DFS0566I message prints, the command was ignored because the two required conditions were not satisfied.

Example 12 for /PSTOP Command

Entry ET:

```
/PSTOP REGION 2 AOITOKEN AOITOK2
```

Response ET:

```
DFS058I PSTOP COMMAND IN PROGRESS
```

Response MT:

```
DFS0569I PSTOP OR STOP COMPLETE FOR REGION 2 AOIT AOITOK2.
```

Explanation: If the DFS0569I message prints, the AO application in region 2 waiting for a message for AOI token AOITOK2 is posted; the application receives AIB return code X'00000004' and reason code X'0000004C'.

Example 13 for /PSTOP Command

Entry ET:

```
/PSTOP REGION 2 AOITOKEN  AOITOK2
```

Response ET:

```
 DFS058I PSTOP COMMAND IN PROGRESS
```

Response MTO:

```
DFS1190I REGION 2 NOT WAITING ON AOITOKEN AOITOK2
```

Explanation: If the DFS1190I message prints, the command is ignored because region 2 was not waiting on AOI token AOITOK2.

Example 14 for /PSTOP Command

Entry ET:

```
/DIS ACTIVE REG
```

Response ET:

```
REGID JOBNAME   TYPE  TRAN/STEP PROGRAM  STATUS         CLASS
    1 MPP610C   TP    NQF1      PMVAPZ12 ACTIVE           1,  2,  3,  4
      BATCHREG  BMP   NONE
      FPRGN     FP    NONE
      DBTRGN    DBT   NONE
      DBRECTA9  DBRC
      DLIECTA9  DLS
      *96081/150611*
```

Explanation: Message processing program PMVAPZ12 is processing transaction NQF1. The job name of the region is MPP610C

Entry ET:

```
/PSTOP REGION JOBNAME MPP610C TRAN NQF1
```

Response ET:

```
DFS058I PSTOP COMMAND IN PROGRESS
DFS0569I PSTOP OR STOP COMPLETE FOR REGION 00001 TRAN NQF1
```

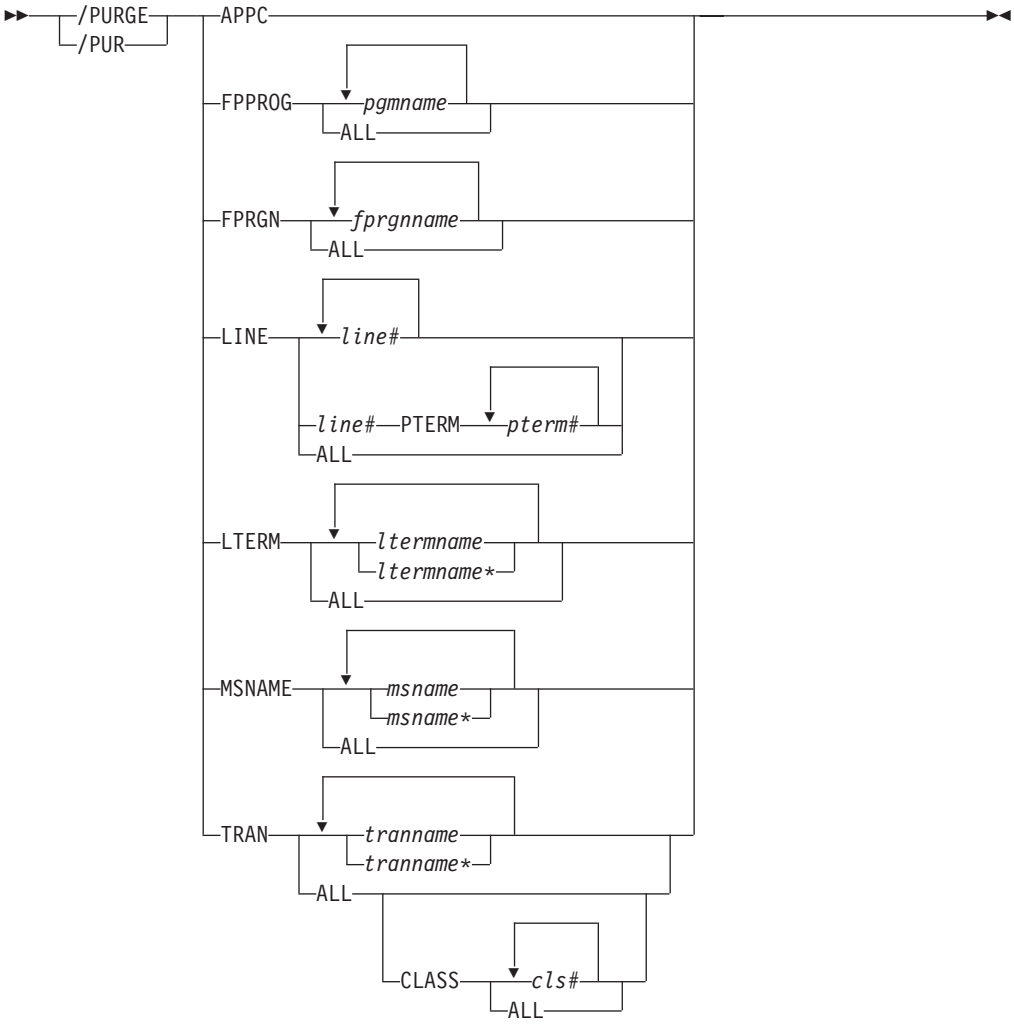
Response ET:

```
DFS058I PSTOP COMMAND IN PROGRESS
DFS0566I PSTOP NOT VALID FOR TRAN NQF1
```

Explanation: If the DFS0569I message is displayed, processing if the transaction type, NQF1, is stopped in message region 1. If the DFS0566I message is displayed, the command was ignored because the two required conditions were not satisfied.

Chapter 41. /PURGE

Format



Environments and Keywords

Table 84 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 84. Valid Environments for the /PURGE Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/PURGE	X		X
APPC	X		X
CLASS	X		X
FPPROG	X		X
FPRGN	X		X
LINE	X		X

© Copyright IBM Corp. 1974, 2003

Table 84. Valid Environments for the /PURGE Command and Keywords (continued)

Command / Keywords	DB/DC	DBCTL	DCCTL
LTERM	X		X
MSNAME	X		X
PTERM	X		X
TRAN	X		X

Usage

/PURGE stops input for a particular communication line, terminal, or logical link path, or stops input messages destined for a particular transaction code. Messages can be sent to the specified communication line or terminal, and transactions can still be scheduled.

/PURGE validity checks all parameters entered by the terminal operator. If an error is detected on parameters that are independent of one another, only the invalid parameters are indicated as being in error and the /PURGE command processes the rest of the parameters.

/PURGE can be used to reset conditions previously set by the /START, /RSTART, /STOP, /PSTOP, or /MONITOR command.

APPC
Is used to purge incoming transactions. All new requests by APPC/z/OS to schedule a transaction in IMS are rejected with TP_Not_Available_No_Retry. Transactions that IMS has already received are processed normally. Sending of output to LU 6.2 devices proceeds normally. Because /PURGE APPC does not call to APPC/z/OS[1], the rejection of transaction scheduling is done by IMS's schedule exit only.

The /PURGE APPC command sets the PURGING status and resets conditions previously set by the /START APPC command. The command is rejected if the APPC is already in DISABLED, FAILED, STOPPED, or CANCEL state.

FPPROG
Specifies the PSB name of the message-driven program to be terminated.

/PURGE takes message-driven programs out of wait-for-input mode and terminates them as soon as their load balancing group message queue is empty.

Use the FPPROG keyword with caution, because all Fast Path message-driven programs that are using a PSB with the same name will be terminated.

FPREGION
Specifies the region identifier of the message-driven program to be terminated.

LINE
Specifies the communication line for which input is to be stopped.

LTERM
Specifies the logical terminal for which input is to be stopped.

1. The sense code returned to the LU 6.2 remote device for an incoming ATTACH to a purged APPC/IMS system is determined by APPC/z/OS, and it might differ from release to release. In general, the remote LU 6.2 application should wait for a period of time after rejection before any attempts to reestablish a session with IMS.

The /PURGE LTERM command is rejected for LTERMs in QLOCK state. (QLOCK indicates that the LTERM is locked from sending any further output or from receiving input that can create additional output for the same LTERM until the state is reset by a specific request received on the session.) /PURGE LTERM is also rejected for remote logical terminals. The LTERM supports generic parameters where the generic parameter specifies LTERMs that already exist.

The /PURGE LTERM command is valid only for LTERMs that belong to nodes that are logged on.

MSNAME
Specifies the logical link path in a multiple systems configuration for which input is to be stopped. The MSNAME keyword supports generic parameters.

TRANSACTION
Specifies the transaction code for which input messages are to be stopped.

The TRANSACTION parameter can be generic where the generic parameter specifies transactions that already exist.

Examples

Example 1 for /PURGE Command

Entry ET:

```
/PURGE FPPROG ALL
```

Response ET:

```
DFS058I  PURGE COMMAND COMPLETED
```

Explanation: All message-driven programs are taken out of wait-for-input mode and terminated by PSB name as soon as their load balancing group message queue is empty.

Example 2 for /PURGE Command

Entry ET:

```
/PURGE FPREGION ALL
```

Response ET:

```
DFS058I  PURGE COMMAND COMPLETED
```

Explanation: All message-driven programs are taken out of wait-for-input mode and terminated by region identifier as soon as their load balancing group message queue is empty.

Example 3 for /PURGE Command

Entry ET:

```
/PURGE LINE 4
```

Response ET:

```
DFS058I PURGE COMMAND COMPLETED
```

Response RT:

```
DFS059I  TERMINAL PURGING
```

Explanation: All physical terminals associated with line 4 can receive output sent to them but are not allowed to enter input.

Example 4 for /PURGE Command

Entry ET:

```
/PURGE LINE 5 7 400
```

Response ET:

```
DFS058I  PURGE COMMAND COMPLETED EXCEPT LINE 400
```

Explanation: All physical terminals associated with line 5 and line 7 can receive output but are not allowed to enter input. Line 400 is an invalid line number.

Example 5 for /PURGE Command

Entry ET:

```
/PURGE MSNAME BOSTON
```

Response ET:

```
DFS058I  PURGE COMMAND COMPLETED
```

Explanation: All messages from a terminal (primary requests), except messages continuing a conversation, will not be queued for the destinations represented by MSNAME BOSTON. This includes all messages destined for remote transactions with the SYSID of the MSNAME, and for remote logical terminals associated with this MSNAME.

Example 6 for /PURGE Command

Entry ET:

```
/PURGE TRANSACTION PIT, SEED
```

Response ET:

```
DFS058I  PURGE COMMAND COMPLETED
```

Explanation: Transactions PIT and SEED can still be scheduled but input for these transactions cannot be queued unless the input originates as output from an application program.

Example 7 for /PURGE Command

Entry ET:

```
/PURGE TRANSACTION ALL CLASS 2
```

Response ET:

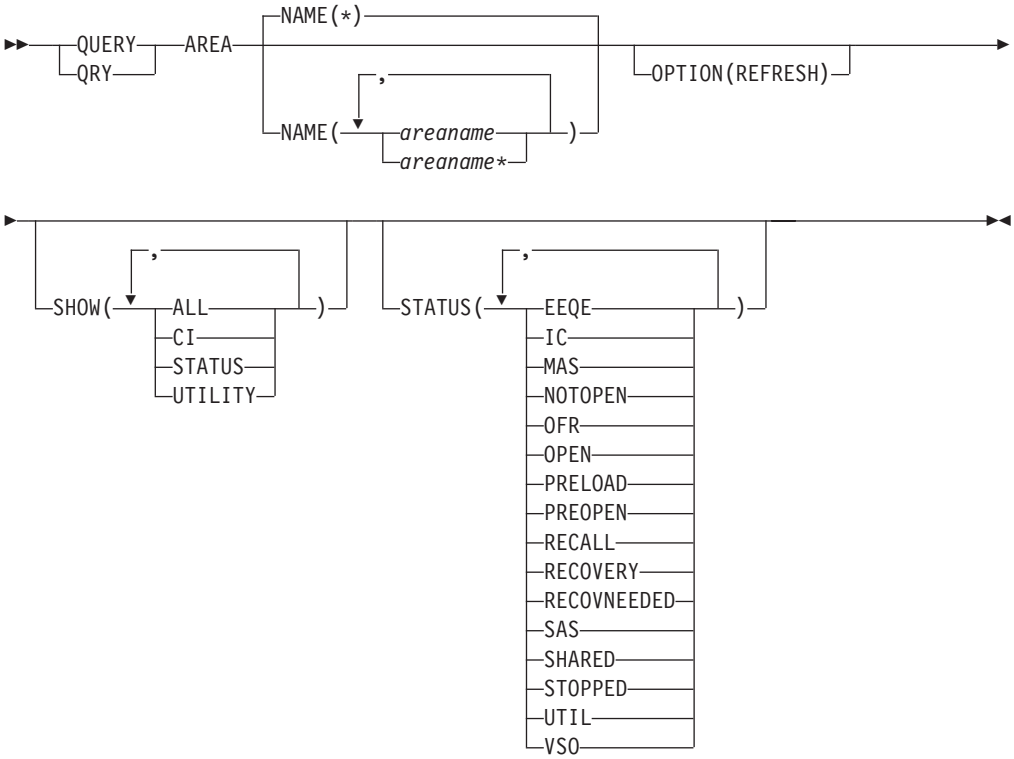
```
DFS058I  PURGE COMMAND COMPLETED
```

Explanation: All transactions associated with class 2 are marked as purged. No further transactions are queued from terminals.

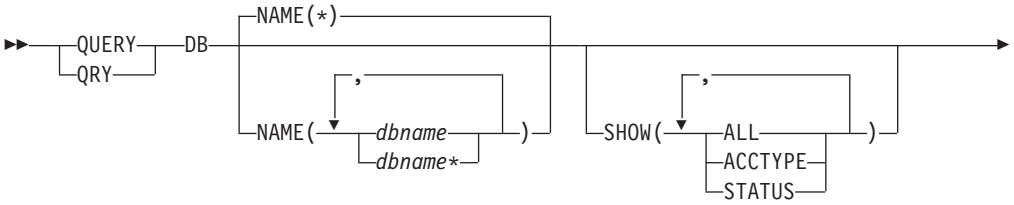
Chapter 42. QUERY

Format

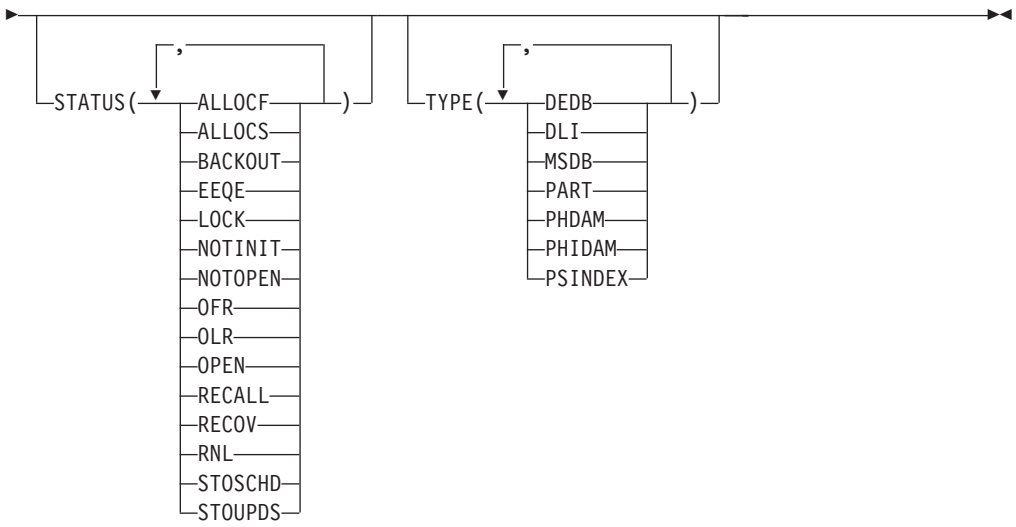
QUERY AREA



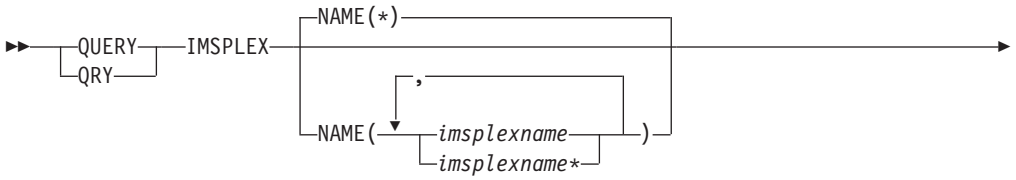
QUERY DB

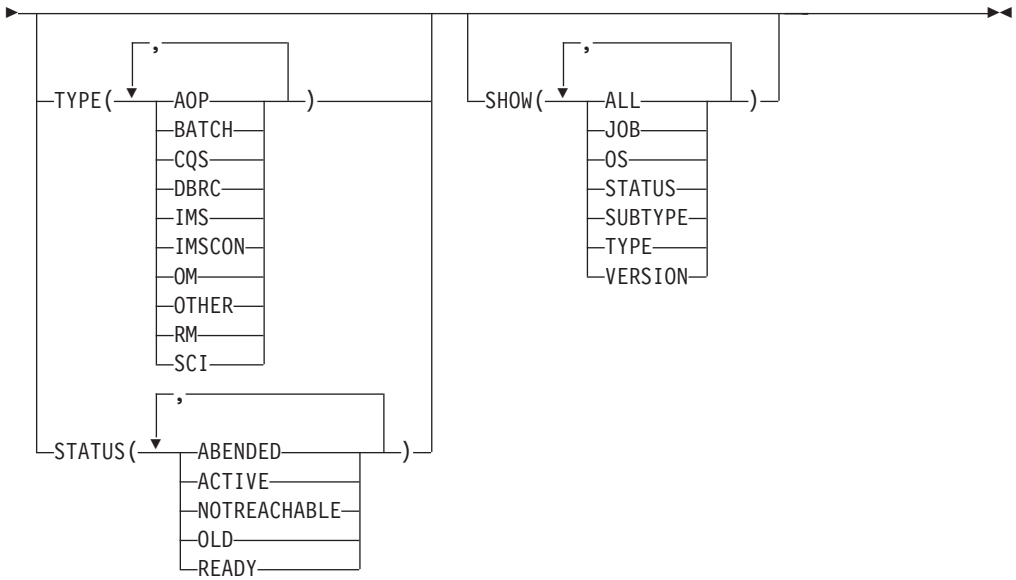


© Copyright IBM Corp. 1974, 2003

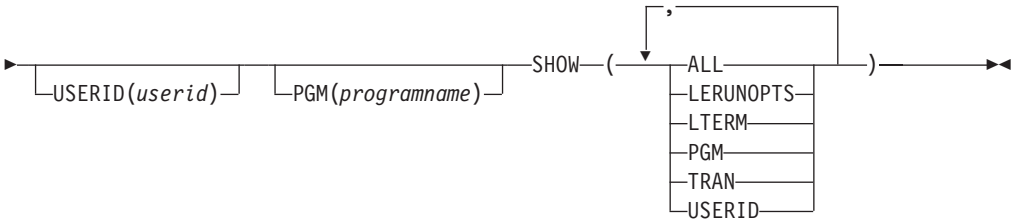


QUERY IMSPLEX

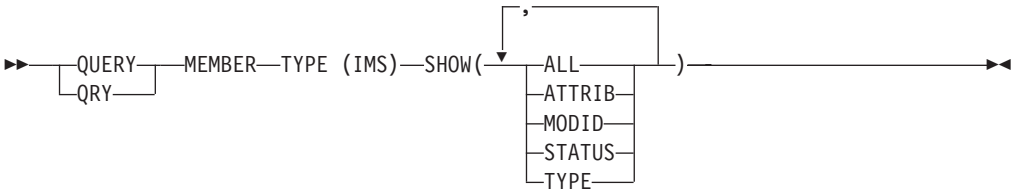




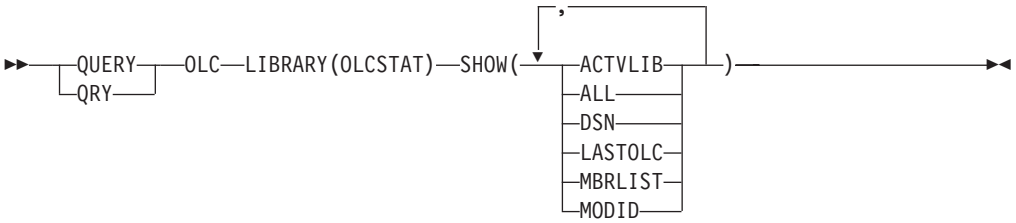
QUERY LE



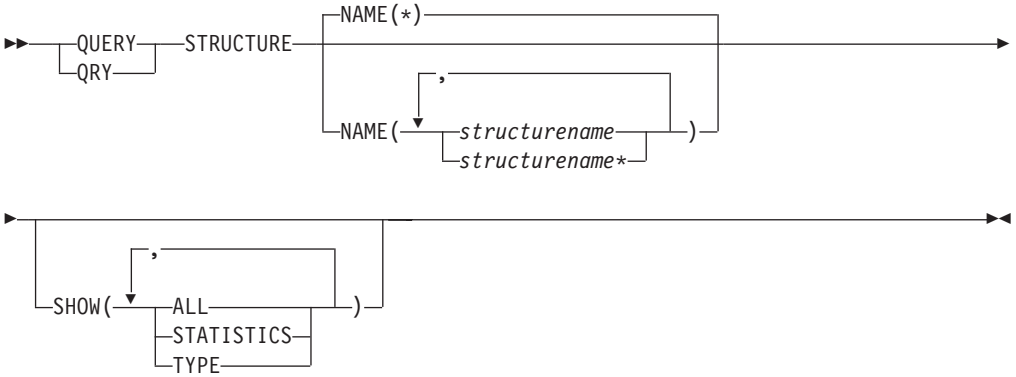
QUERY MEMBER



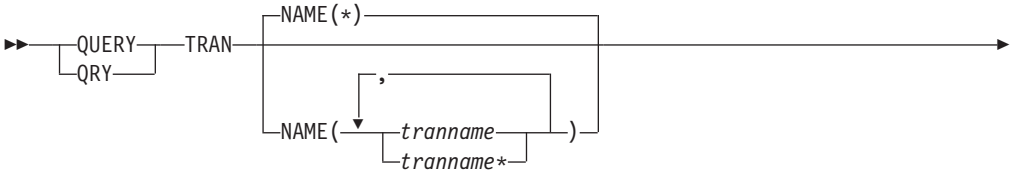
QUERY OLC

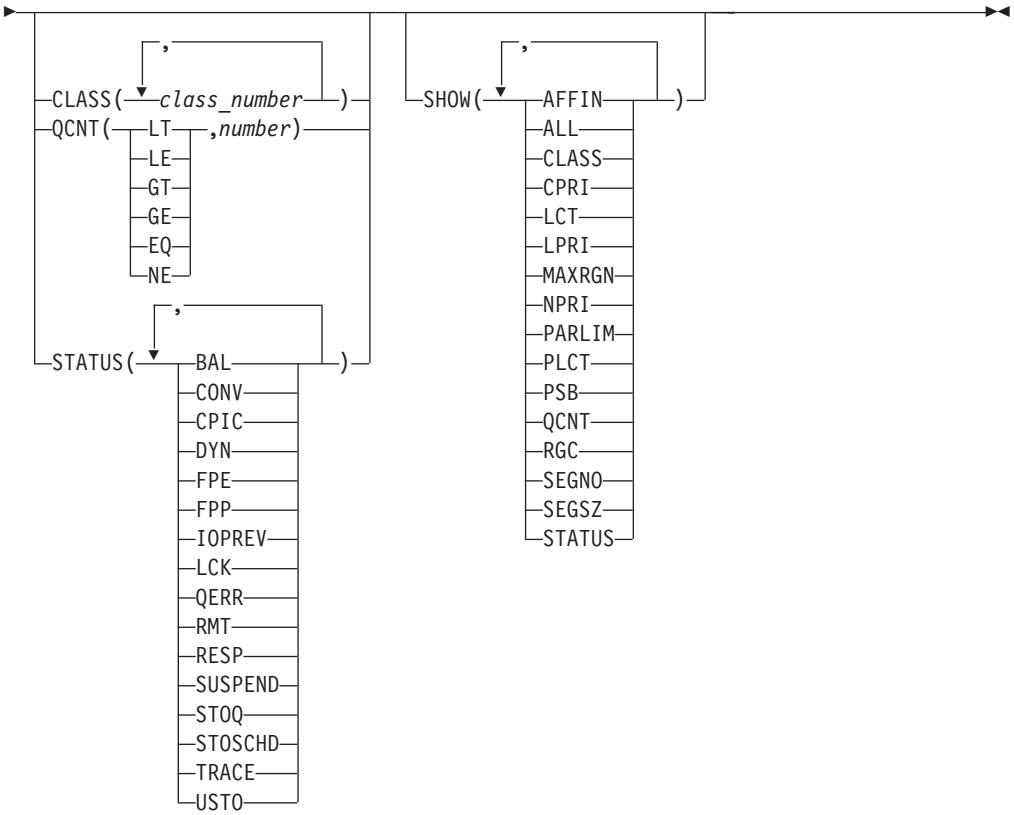


QUERY STRUCTURE



QUERY TRAN





Environments and Keywords

Table 85, Table 86, Table 87 on page 469, Table 88 on page 469, Table 89 on page 469, Table 90 on page 469, Table 91 on page 469, and Table 92 on page 470 list the environments (DB/DC, DBCTL, and DCCTL) from which the QUERY command and keywords can be issued.

Table 85. Valid Environments for QUERY AREA Command and Keywords

Command / Keyword	DB/DC	DBCTL	DCCTL
QUERY AREA	X	X	
NAME	X	X	
OPTION	X	X	
SHOW	X	X	
STATUS	X	X	

Table 86. Valid Environments for QUERY DB Command and Keywords

Command / Keyword	DB/DC	DBCTL	DCCTL
QUERY DB	X	X	
NAME	X	X	
SHOW	X	X	
STATUS	X	X	
TYPE	X	X	

Table 87. Valid Environments for QUERY IMSPLEX Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
QUERY IMSPLEX [1]			
NAME			
SHOW			
STATUS			
TYPE			

Note:

1. There are no environment indicators for the `QUERY IMSPLEX` command itself because it does not run in any IMS control or dependent region's address space. `QUERY IMSPLEX` is processed in an OM command processing environment.

Table 88. Valid Environments for the QUERY LE Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
QUERY LE	X	X	X
LTERM	X	X	X
PGM	X	X	X
SHOW	X	X	X
TRAN	X	X	X
USERID	X	X	X

Table 89. Valid Environments for the QUERY MEMBER Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
QUERY MEMBER	X	X	X
ALL	X	X	X
ATTRIB	X	X	X
SHOW	X	X	X
STATUS	X	X	X
TYPE	X	X	X

Table 90. Valid Environments for the QUERY OLC Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
QUERY OLC	X	X	X
LIBRARY	X	X	X
SHOW	X	X	X

Table 91. Valid Environments for the QUERY STRUCTURE Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
QUERY STRUCTURE [1]			
ALL			
NAME			
SHOW			

Table 91. Valid Environments for the QUERY STRUCTURE Command and Keywords (continued)

Command / Keywords	DB/DC	DBCTL	DCCTL
STATISTICS			
TYPE			

Note:

1. There are no environment indicators for the QUERY STRUCTURE command itself because it does not run in any IMS control or dependent region's address space. QUERY STRUCTURE is processed in an RM command processing environment.

Table 92. Valid Environments for the QUERY TRAN Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
QUERY TRAN	X		X
CLASS	X		X
NAME	X		X
QCNT	X		X
SHOW	X		X
STATUS	X		X

QUERY AREA

QUERY AREA is an IMSplex command that displays information about DEDB areas and area data set information. The command can only be specified through the Operations Manager (OM) API and can only be processed by DB/DC and DBCTL environments. In addition, the QUERY AREA command is valid on the XRF alternate as well as the RSR tracker.

The command syntax for this command is defined in XML and is available to automation programs which communicate with OM.

NAME()
Specifies the names of the specific areas that are to be processed or the group of areas whose names match the generic or wildcard parameter specified.

If the STATUS filter is not specified, all the area names that match the NAME parameter are returned. The NAME keyword is optional and the default is NAME(*).

OPTION()
Specifies the additional functions to be performed.

The filters supported with the OPTION keyword are:

REFRESH
Refreshes the control interval information for the sequential dependent space and the independent overflow part of the direct addressable space for the area. OPTION(REFRESH) returns the CI information even if SHOW(CI) is not specified. OPTION(REFRESH) results in an I/O request that is performed only at the command master IMS if the area is open at the master. If the command is routed to multiple IMSs in the IMSplex, all non-master IMSs return

local CI information. The age of the local information shown on the non-master IMS systems is at most as old as the value set on the IOVFI=IMS control region parameter.

The CI information is returned only if the area is open at the IMS. If the area is not open, blanks will be returned in the output CI columns SDAT, SDAU, LDAT, and LDAU. If the area is not open at the command master IMS, no refresh of the control intervals is performed. The `QRY AREA NAME(areaname) SHOW(CI) OPTION(REFRESH)` must be routed to the IMS where the area is open to get the current SDEP and IOVF CI information.

If the default NAME(*) is used with the REFRESH keyword, or if large numbers of areas are processed with the REFRESH keyword, performance may be affected depending on the size and number of areas involved. If large areas, numerous areas, or both are involved, the control regions processing the command may appear stopped.

SHOW()
Specifies the area output fields to be returned. The area name, the DEDB name, and the area data set information are always returned along with the name of the IMS that created the output for the area and the completion code.

The filters supported with the SHOW keyword are:

ALL Returns all the output fields.

CI Control intervals.

Returns the total and unused control intervals defined for the sequential dependent space and the total and unused control intervals for the independent overflow part of the direct addressable space.

STATUS
Local area status.

If the area has an EEQE status, the count of I/O errors or write error EEQE for the area are also returned.

UTILITY
Returns the utility information about the utility that has the area open. The utility name, the total and available buffers in the private pool, and the utility UOW are returned.

STATUS()
Selects areas for display that match the NAME parameter and possess at least one of the specified area status.

This allows for additional filtering by area status. The output returned when the STATUS filter is specified includes the status of the area that caused the area name to be displayed even if the `SHOW(STATUS)` is not specified.

Status parameters are the same as the value displayed in the `QUERY AREA` output.

QUERY AREA Output Fields

Table 93 on page 472 shows the `QUERY AREA` output fields. The columns in the table are as follows:

Short Label
Contains the short label generated in the XML output.

Keyword
Identifies the keyword on the command that caused the field to be generated. N/A appears for output fields that are always returned.

Scope Identifies the scope of the output field.

Meaning
Provides a brief description of the output field.

Table 93. Output Fields for QUERY AREA Command

Short Label	Keyword	Scope	Meaning
ADS	N/A	N/A	ADS name The Area data set name or names associated with the AREA.
AREA	N/A	N/A	Area name The Area name is always returned.
CC	N/A	N/A	Completion code The completion code indicates whether or not IMS was able to process the command for the specified resource. The completion code is always returned. See the return, reason, and completion codes table for QUERY AREA.
DB	N/A	N/A	DEDB name The DEDB name associated with the Area.
LDAT	CI	LCL	Local value of the total control intervals for the independent overflow part of the direct addressable space. This value only appears if the area is open.
LDAU	CI	LCL	Local value of the unused control intervals for the independent overflow part of the direct addressable space. This value only appears if the area is open and the IOVF count ITASK was not disabled when IOVFI=1 on the IMS Control Region startup procedure was specified. When the command is processed on an RSR tracking IMS system, no information will be returned. This value is refreshed during command processing if OPTION(IOVF) was entered on the QUERY AREA command. If OPTION(IOVF) was not specified, the value reflects the updated value from the last IOVF count ITASK.
LEQ	STATUS	LCL	Local value of the total control intervals for the independent overflow part of the direct addressable space. This value only appears if the area is open.
LPBA	UTILITY	LCL	Available number of private buffers in private pool.
LPBT	UTILITY	LCL	Total number of private buffers in private pool.

Table 93. Output Fields for QUERY AREA Command (continued)

Short Label	Keyword	Scope	Meaning
LSDT	CI	LCL	Local value of the total control intervals defined for the sequential dependent space. This value only appears if the area is open and SDEPs have been defined.
LSDU	CI	LCL	Local value of the unused control intervals defined for the sequential dependent space. This value only appears if the area is open, SDEPs have been defined, and unused SDEPs are available.
LSTT	STATUS	LCL	Local area status All area status conditions that apply are returned.
LUOW	UTILITY	LCL	The current utility UOW for HSREORG and HSSP, otherwise it is blank.
LUTIL	UTILITY	LCL	Utility name that has area OPEN.
MBR	N/A	N/A	IMSplex member that built the output line IMS identifier of the IMS that built the output. The IMS identifier is always returned.

QUERY AREA STATUS

Table 94 lists the database status conditions that may be returned when SHOW(STATUS) is specified.

Table 94. Status Conditions for QUERY AREA

Status	Meaning
EEQE	Area has EEQEs.
IC	Area image copy is active.
MAS	Area is on a multi-area structure.
NOTOPEN	Area is not open.
OFR	Area has online forward recovery in progress to bring it up to current tracking level.
OPEN	Area is open.
PRELOAD	Area is defined to be preloaded.
PREOPEN	Area is defined to be preopened.
RECALL	Area is in recall.
RECOVERY	Area recovery in progress.
RECOVNEEDED	Area needs recovery.
SAS	Area is on a single area structure.
SHARED	Area is shared.
STOPPED	Area is stopped.
UTIL	Area is open by a utility.
VSO	Area is a VSO area.

The following table, Table 95 on page 474, lists the ADS status conditions that can be returned for an ADS associated with an AREA when SHOW(STATUS) is specified.

Table 95. ADS Status Conditions for QUERY AREA

Status	Meaning
COPY-PHASE	The CREATE utility is active on this ADS and is in the COPY phase. The CREATE utility must complete before any action can be processed for the ADS.
FORMAT-PHASE	The CREATE utility is active on this ADS and is in the FORMAT phase. The CREATE utility must complete before any action can be processed for the ADS.
LONGBUSY	Area in long busy state or long busy recovery mode.
PREOPEN-FAIL	XRF PREOPEN failed for this ADS.
SEVERE-ERROR	The ADS had a severe I/O error (write error to 2nd CI).
UNAVAIL	The ADS is marked unavailable due to I/O errors.

Return, Reason, and Completion Codes for the QUERY AREA Command

An IMS return and reason code is returned to OM by the QUERY AREA command. The OM return and reason codes that may be returned as a result of the QUERY AREA command are standard for all commands entered through the OM API. See *IMS Version 9: Common Service Layer Guide and Reference* for a list of the OM codes and the code meanings.

Table 96 includes the return and reason codes and a brief explanation of the QUERY AREA command. Table 97 on page 475 includes an explanation of the completion codes. Errors unique to the processing of QUERY AREA command are returned as completion codes. A completion code is returned for each action against an individual resource.

Table 96. Return and Reason Code for the QUERY AREA Command

Return Code	Reason Code	Meaning
X'00000000'	X'00000000'	The QUERY AREA command completed successfully.
X'00000008'	X'0000200C'	The QUERY AREA command is not processed because no resources matched any status specified on the STATUS( ) keyword.
X'00000008'	X'00002014'	The QUERY AREA command is not processed because in invalid character is found in the area name parameter.
X'00000008'	X'00002040'	More than one filter or keyword value is specified on the QUERY AREA command. Either more than one keyword or an invalid combination of filters was specified. Check the input command and reenter the correct combinations.
X'0000000C'	X'00003000'	The QUERY AREA command was successful for at least one resource name. The QUERY AREA command was not successful for one or more resource names. The completion code indicates the reason for the error with the resource name. The completion codes that can be returned by the QUERY AREA command are listed in the QUERY AREA Completion Code table.

Table 96. Return and Reason Code for the QUERY AREA Command (continued)

Return Code	Reason Code	Meaning
X'0000000C'	X'00003004'	The QUERY AREA command was not successful for all the resource name(s) specified. The completion code indicates the reason for the error with the resource name. The completion codes that can be returned by the QUERY AREA command are listed in the QUERY AREA completion code table.
X'00000010'	X'00004024'	The QUERY AREA command cannot be processed on a non-Fast Path system.
X'00000014'	X'00005004'	The QUERY AREA command processing terminated as a DFSOCMD response buffer could not be obtained.
X'00000014'	X'00005FFF'	The QUERY AREA command processing terminated due to an internal error.

Table 97. Completion Codes for the QUERY AREA Command

Completion Code	Meaning
0	The QUERY AREA command completed successfully for the resource.
10	Resource not found The resource name is unknown to the client that is processing the request. The resource name may have been typed in error or the resource may not be active at this time. Confirm that the correct spelling of the resource name is specified on the command.

QUERY AREA Compared to Other Commands

Table 98 shows different instances of the QUERY AREA command and other IMS commands that perform similar functions.

Table 98. QUERY AREA Compared to Similar IMS Commands

QUERY AREA Command	SImilar IMS Command
QUERY AREA	/DIS AREA area1...arean | ALL, /DIS STATUS AREA

Examples for QUERY AREA Command

This section provides OM API and TSO SPOC input and output examples and explanations for the QUERY command.

Example 1 for QUERY AREA

TSO SPOC input:

```
QRY AREA NAME(DB21AR10) SHOW(STATUS,CI)
```

TSO SPOC output:

```
AreaName ADSName  MbrName  DBName     CC  SDep-T  SDep-U   Dir-T Dir-U EQCnt LclStat
DB21AR0           IMS2     DEDBJN21    0                                    PREOPEN,NOTOPEN
DB21AR0           SYS3     DEDBJN21    0    1303    1302   74    74          PREOPEN,OPEN,SHARED
DB21AR0  DB21AR01 SYS3                 0                                10
DB21AR0  DB21AR02 SYS3                 0                                10
```

OM API input:

```
CMD(QRY AREA NAME(DB21AR10) SHOW(STATUS,CI))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.2.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2003.132 16:10:52.861123</statime>
<stotime>2003.132 16:10:52.862301</stotime>
<staseq>B968A1B61BEC302F</staseq>
<stoseq>B968A1B61C35D38E</stoseq>
<rqsttkn1>USRT005 10091052</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<master>IMS2    </master>
<userid>USRT005 </userid>
<verb>QRY </verb>
<kwd>AREA            </kwd>
<input>QRY AREA NAME(DB21AR0) SHOW(CI,STATUS) </input>
</cmd>
<cmdrsphdr>
<hdr slbl="AREA" llbl="AreaName" scope="LCL" sort="a" key="1"
 scroll="no" len="8" dtype="CHAR" align="left" skipb="no" />
<hdr slbl="ADS" llbl="ADSName" scope="LCL" sort="a" key="2" scroll="no"
 len="8" dtype="CHAR" align="left" skipb="yes" />
<hdr slbl="MBR" llbl="MbrName" scope="LCL" sort="a" key="3" scroll="no"
 len="8" dtype="CHAR" align="left" skipb="no" />
<hdr slbl="DB" llbl="DBName" scope="LCL" sort="n" key="0" scroll="no"
 len="8" dtype="CHAR" align="left" skipb="no" />
<hdr slbl="CC" llbl="CC" scope="LCL" sort="n" key="0" scroll="yes"
 len="4" dtype="INT" align="right" skipb="no" />
<hdr slbl="LSDT" llbl="SDep-T" scope="LCL" sort="n" key="0"
 scroll="yes" len="7" dtype="INT" align="right" skipb="no" />
<hdr slbl="LSDU" llbl="SDep-U" scope="LCL" sort="n" key="0"
 scroll="yes" len="7" dtype="INT" align="right" skipb="no" />
<hdr slbl="LDAT" llbl="Dir-T" scope="LCL" sort="n" key="0" scroll="yes"
 len="7" dtype="INT" align="right" skipb="no" />
<hdr slbl="LDAU" llbl="Dir-U" scope="LCL" sort="n" key="0" scroll="yes"
 len="7" dtype="INT" align="right" skipb="no" />
<hdr slbl="LEQ" llbl="EQCnt" scope="LCL" sort="n" key="0" scroll="yes"
 len="5" dtype="INT" align="right" skipb="yes" />
<hdr slbl="LSTT" llbl="LclStat" scope="LCL" sort="n" key="0"
 scroll="yes" len="*" dtype="CHAR" align="left" skipb="no" />
</cmdrsphdr>
<cmdrspdata>
<rsp>AREA(DB21AR0 ) MBR(IMS2    ) DB(DEDBJN21) CC(   0) LSDT( ) LSDU( )
 LDAT( ) LDAU( ) LSTT(PREOPEN,NOTOPEN) </rsp>
<rsp>AREA(DB21AR0 ) MBR(SYS3    ) DB(DEDBJN21) CC(   0) LSDT(   1303)
 LSDU(   1302) LDAT(     74) LDAU(     74) LSTT(PREOPEN,OPEN,SHARED)
</rsp>
<rsp>AREA(DB21AR0 ) ADS(DB21AR01) MBR(SYS3    ) DB( ) CC(   0) LEQ(  10) </rsp>
<rsp>AREA(DB21AR0 ) ADS(DB21AR02) MBR(SYS3    ) DB( ) CC(   0) LEQ(  10) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: The command returns the CI and STATUS information for the area, DB21AR0, from all the IMSs in the IMSPlex. Any ADS information, if available, is also returned by each IMS.

Example 2 for QUERY AREA Command

TSO SPOC input:

```
QRY AREA NAME(DB21AR1*) STATUS(SHARED)
```

TSO SPOC output:

```
Log for . . : QRY AREA NAME(DB21AR1*) STATUS(SHARED)
IMSplex . . . . . : PLEX1
Routing . . . . . :
Start time. . . . : 2003.132 09:13:37.93
Stop time . . . . : 2003.132 09:13:37.94
Return code . . . : 0200000C
Reason code . . . : 00003000
Command master. . : IMS2

                Return     Reason
MbrName     Code       Code
--------    ------     ------
IMS2        00000008   0000200C

AreaName ADSName  MbrName  DBName     CC EQCnt LclStat
DB21AR10          SYS3     DEDBJN21    0       PREOPEN,OPEN,SHARED
DB21AR10 DB21AR10 SYS3                 0    10
DB21AR11          SYS3     DEDBJN21    0       PREOPEN,OPEN,SHARED
DB21AR11 DB21AR11 SYS3                 0    10
```

OM API input:

```
CMD(QRY AREA NAME(DB21AR1*) STATUS(SHARED))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.2.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2003.132 16:13:37.940282</statime>
<stotime>2003.132 16:13:37.940959</stotime>
<staseq>B968A2538A73A707</staseq>
<stoseq>B968A2538A9DF027</stoseq>
<rqsttkn1>USRT005 10091337</rqsttkn1>
<rc>0200000C</rc>
<rsn>00003000</rsn>
</ctl>
<cmderr>
<mbr name="IMS2    ">
<typ>IMS     </typ>
<styp>DBDC    </styp>
<rc>00000008</rc>
<rsn>0000200C</rsn>
</mbr>
</cmderr>
<cmd>
<master>IMS2    </master>
<userid>USRT005 </userid>
<verb>QRY </verb>
<kwd>AREA            </kwd>
<input>QRY AREA NAME(DB21AR1*) STATUS(SHARED) </input>
</cmd>
<cmdrsphdr>
<hdr slbl="AREA" llbl="AreaName" scope="LCL" sort="a" key="1"
 scroll="no" len="8" dtype="CHAR" align="left" skipb="no" />
<hdr slbl="ADS" llbl="ADSName" scope="LCL" sort="a" key="2" scroll="no"
 len="8" dtype="CHAR" align="left" skipb="yes" />
<hdr slbl="MBR" llbl="MbrName" scope="LCL" sort="a" key="3" scroll="no"
 len="8" dtype="CHAR" align="left" skipb="no" />
<hdr slbl="DB" llbl="DBName" scope="LCL" sort="n" key="0" scroll="no"
 len="8" dtype="CHAR" align="left" skipb="no" />
<hdr slbl="CC" llbl="CC" scope="LCL" sort="n" key="0" scroll="yes"
 len="4" dtype="INT" align="right" skipb="no" />
<hdr slbl="LEQ" llbl="EQCnt" scope="LCL" sort="n" key="0" scroll="yes"
 len="5" dtype="INT" align="right" skipb="yes" />
<hdr slbl="LSTT" llbl="LclStat" scope="LCL" sort="n" key="0"
```

```
 scroll="yes" len="*" dtype="CHAR" align="left" skipb="no" />
</cmdrsphdr>
<cmdrspdata>
<rsp>AREA(DB21AR10) MBR(SYS3    ) DB(DEDBJN21) CC(   0)
 LSTT(PREOPEN,OPEN,SHARED) </rsp>
<rsp>AREA(DB21AR10) ADS(DB21AR10) MBR(SYS3    ) DB( ) CC(   0) LEQ( 10) </rsp>
<rsp>AREA(DB21AR11) MBR(SYS3    ) DB(DEDBJN21) CC(   0)
 LSTT(PREOPEN,OPEN,SHARED) </rsp>
<rsp>AREA(DB21AR11) ADS(DB21AR11) MBR(SYS3    ) DB( ) CC(   0) LEQ( 10) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: The command returns all the areas that match the wildcard name and have a status of SHARED. The status is also returned. The ADS information for the AREAs is also returned if it is available. Command response lines are not returned from IMS2 because no AREAs match the status specified. A return and reason code is returned from IMS2.

QUERY DB

QUERY DB is an IMSplex command that displays information about databases. The command can only be specified through the OM API and can only be processed in DB/DC and DBCTL environments. In addition, QUERY DB is valid on the XRF alternate as well as the RSR tracker.

The command syntax for QUERY DB is defined in XML and is available to automation programs which communicate with OM.

NAME()

Specifies the names of the specific databases that are to be processed or indicates that the command is to be applied to all the databases in the system. NAME(*) is the default.

If the STATUS filter is not specified, all the database names that match the NAME parameter are returned.

Wildcard parameters can be specified. The database names that match the generic or wildcard parameters are processed. Response lines are returned for all the databases names that are processed.

The database name specified can be a HALDB master or a HALDB partition. If the database name is the HALDB master, response lines are returned for the HALDB master and all of its partitions. If the database name is the HALDB partition, response lines are returned for the HALDB master and the partition name if the HALDB master has not been taken offline.

If the database name specified is a DEDB name, response lines are returned for the DEDB name and all the DEDB areas.

SHOW()

Specifies the database output fields to be returned. The database name and type are always returned along with the name of the IMS that created the output for the database and the completion code. The filters supported with the SHOW keyword are:

ALL Returns all the output fields.

ACCTYP

Type of access to database which can be one of the following:

- BRWS - Read only

- EXCL - Exclusive
- READ - Read
- UPD - Update

STATUS
Local database status.

STATUS()
Selects databases for display that match the NAME parameter and possess at least one of the specified database status. This allows for additional filtering by database status.

The output returned when the STATUS filter is specified includes the status of the database that caused the database name to be displayed even if the SHOW(STATUS) is not specified.

Status parameters are the same as the value displayed in the QUERY DB output.

TYPE()
Selects databases for display that match the NAME parameter and specific TYPE filter. The supported TYPE filters are:

- DEDB
- DLI
- MSDB
- PART
- PHDAM
- PHIDAM
- PSINDEX

TYPE can be specified with the STATUS filter. If both TYPE and STATUS filters are specified, a response line is returned for each database that matches the NAME parameter, the TYPE, and STATUS filter specified.

QUERY DB Output Fields

Table 99 shows the QUERY DB output fields. The columns in the table are as follows:

Short Label
Contains the short label generated in the XML output.

Keyword
Identifies the keyword on the command that caused the field to be generated. N/A appears for output fields that are always returned.

Scope Identifies the scope of the output field.

Meaning
Provides a brief description of the output field.

Table 99. Output Fields for QUERY DB Command

Short Label	Keyword	Scope	Meaning
AREA	N/A	N/A	Area name. The Area name is returned if there are one or more response lines for DEDB areas in the output.

Table 99. Output Fields for QUERY DB Command (continued)

Short Label	Keyword	Scope	Meaning
CC	N/A	N/A	Completion code. The completion code indicates whether or not IMS was able to process the command for the specified resource. The completion code is always returned. Refer to the return, reason, and completion codes for QUERY DB.
DB	N/A	N/A	Database name. The database name is always returned.
LACC	ACCTYP	LCL	Type of access to database which can be one of the following: • BRWS - Read only • EXCL - Exclusive • READ - Read • UPD - Update
LSTT	STATUS	LCL	Local database status. All database status conditions that apply are returned.
MBR	N/A	N/A	IMSplex member that built the output line. The IMS identifier of the IMS that built the output. The IMS identifier is always returned.
PART	N/A	N/A	HALDB partition name. The partition name is returned if there is one or more response lines for HALDB partitions in the output.
TYP	N/A	N/A	The type of the database. • AREA - indicates the response line is for a DEDB area • blank - if database status is NOTINIT • DEDB - indicates the database is a DEDB • DL/I - indicates the database is a full function non-partitioned database • MSNR - indicates the database is an MSDB non-related database • MSRD - indicates the database is an MSDB related dynamic database • MSRF - indicates the database is an MSDB related fixed database • PART - indicates the database is a HALDB partition • PHDAM - indicates the database is the master of a partitioned HDAM database • PHIDAM - indicates the database is the master of a Partitioned HIDAM database • PSINDEX - indicates the database is the master of a partitioned secondary index database

QUERY DB Status

Table 100 on page 481 lists the database status conditions that may be returned when SHOW(STATUS) is specified.

Table 100. Database Status Conditions for the QUERY DB Command

Status	Meaning
ALLOCF	Database has an allocation failure.
ALLOCS	Database is allocated successfully.
BACKOUT	Incomplete backout exists for the database which prevents the use of the database.
EEQE	One or more extended error queue elements exist for the database.
LOCK	Database is locked.
NOTINIT	Database directory initialization failed.
NOTOPEN	Database is not open.
OFR	Database has online forward recovery in progress to bring it up to current tracking level.
OLR	Database partition has Online Reorganization in progress locally.
OPEN	Database is open.
RECALL	Database recall is in progress.
RECOV	Database recovery is in progress.
RNL	Randomizer not loaded for the DEDB database.
STOSCHD	Database scheduling is stopped.
STOUPDS	Database updates are stopped because an `UPDATE DB STOP(UPDATES)` command or a `/DBDUMP DATABASE` command was issued.

Return, Reason, and Completion Codes for the QUERY DB Command

An IMS return and reason code is returned to OM by the `QUERY DB` command. The OM return and reason codes that may be returned as a result of the `QUERY DB` command are standard for all commands entered through the OM API. See *IMS Version 9: Common Service Layer Guide and Reference* for a list of the OM codes and the code meanings.

Table 101 includes the return and reason codes and a brief explanation of the `QUERY DB` command. Table 102 on page 482 includes an explanation of the completion codes. Errors unique to the processing of `QUERY DB` command are returned as completion codes. A completion code is returned for each action against an individual resource.

Table 101. Return and Reason Codes for the QUERY DB Command

Return Code	Reason Code	Meaning
X'00000000'	X'00000000'	The `QUERY DB` command completed successfully.
X'00000004'	X'00001010'	The `QUERY DB` command is not processed because no resources were found that matched the NAME parameter and the STATUS or TYPE filter.
X'00000008'	X'00002014'	The `QUERY DB` command is not processed because an invalid character is found in the database name parameter.
X'00000008'	X'00002040'	More than one filter or keyword value is specified on the `QUERY DB` command. Either more than one keyword or an invalid combination of filters was specified. Check the input command and reenter the correct combinations.

Table 101. Return and Reason Codes for the QUERY DB Command (continued)

Return Code	Reason Code	Meaning
X'0000000C'	X'00003000'	The QUERY DB command was successful for at least one resource name. The QUERY DB command was not successful for one or more resource names. The completion code indicates the reason for the error with the resource name. The completion codes that can be returned by the QUERY DB command are listed in the QUERY DB completion code table.
X'0000000C'	X'00003004'	The QUERY DB command was not successful for all the resource name(s) specified. The completion code indicates the reason for the error with the resource name. The completion codes that can be returned by the QUERY DB command are listed in the QUERY DB completion code table.
X'00000014'	X'00005004'	The QUERY DB command processing terminated as a DFSOCMD response buffer could not be obtained.
X'00000014'	X'00005FFF'	The QUERY DB command processing terminated due to an internal error.

Table 102. Completion Codes for the QUERY DB Command

Completion Code	Meaning
0	The QUERY DB command completed successfully for the resource.
10	Resource not found The resource name is unknown to the client that is processing the request. The resource name may have been typed in error or the resource may not be active at this time. Confirm the correct spelling of the resource name specified on the command.

QUERY DB Compared to Other Commands

Table 103 shows different instances of the QUERY DB command and other IMS commands that perform similar functions.

Table 103. QUERY DB Command Compared to Similar Commands

QUERY DB Command	Similar IMS Commands
QUERY DB	/DIS DB dbname1...dbnamen | ALL, /DIS STATUS DB

Examples for QUERY DB Command

This section provides OM API and TSO SPOC input and output examples and explanations for the QUERY command.

Example 1 for QUERY DB Command

TSO SPOC input:

```
QRY DB NAME(BE3PARTS,DEDBJN21,DBHDOJ01) SHOW(ALL)
```

TSO SPOC output:

```
DBName   AreaName PartName MbrName    CC TYPE     LAcc LclStat
BE3PARTS                  IMSA        0 DL/I     UPD  NOTOPEN
BE3PARTS                  IMS1        0 DL/I     EXCL NOTOPEN
DBHDOJ01                  IMSA        0 PHDAM    UPD
DBHDOJ01          PDHDOJA IMSA        0 PART     UPD  NOTOPEN
```

```
DBHDOJ01          PDHDOJB  IMSA        0 PART     UPD  NOTOPEN
DBHDOJ01                   IMS1        0 PHDAM    UPD
DBHDOJ01          PDHDOJA  IMS1        0 PART     UPD  NOTOPEN
DBHDOJ01          PDHDOJB  IMS1        0 PART     UPD  NOTOPEN
DEDBJN21                   IMSA        0 DEDB     UPD  NOTOPEN
DEDBJN21 DB21AR0           IMSA        0 AREA          NOTOPEN
DEDBJN21 DB21AR1           IMSA        0 AREA          NOTOPEN
DEDBJN21                   IMS1        0 DEDB     UPD  NOTOPEN
DEDBJN21 DB21AR0           IMS1        0 AREA          NOTOPEN
DEDBJN21 DB21AR1           IMS1        0 AREA          NOTOPEN
```

OM API input:

```
CMD (QRY DB NAME(BE3PARTS,DEDBJN21,DBHDOJ01) SHOW(ALL))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.2.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2003.128 19:20:29.755526</statime>
<stotime>2003.128 19:20:29.756748</stotime>
<staseq>B963C4A21E086F82</staseq>
<stoseq>B963C4A21E54C862</stoseq>
<rqsttkn1>USRT005 10122029</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<master>IMSA    </master>
<userid>USRT005 </userid>
<verb>QRY </verb>
<kwd>DB              </kwd>
<input>QRY DB NAME(BE3PARTS,DEDBJN21,DBHDOJ01) SHOW(ALL) </input>
</cmd>
<cmdrsphdr>
<hdr slbl="DB" llbl="DBName" scope="LCL" sort="a" key="1" scroll="no"
 len="8" dtype="CHAR" align="left" skipb="no" />
<hdr slbl="AREA" llbl="AreaName" scope="LCL" sort="a" key="3"
 scroll="no" len="8" dtype="CHAR" align="left" skipb="yes" />
<hdr slbl="PART" llbl="PartName" scope="LCL" sort="a" key="4"
 scroll="no" len="8" dtype="CHAR" align="left" skipb="yes" />
<hdr slbl="MBR" llbl="MbrName" scope="LCL" sort="a" key="2" scroll="no"
 len="8" dtype="CHAR" align="left" skipb="no" />
<hdr slbl="CC" llbl="CC" scope="LCL" sort="n" key="0" scroll="yes"
 len="4" dtype="INT" align="right" skipb="no" />
<hdr slbl="TYP" llbl="TYPE" scope="LCL" sort="n" key="0" scroll="yes"
 len="8" dtype="CHAR" align="left" skipb="no" />
<hdr slbl="LACC" llbl="LAcc" scope="LCL" sort="n" key="0" scroll="yes"
 len="*" dtype="CHAR" align="left" skipb="no" />
<hdr slbl="LSTT" llbl="LclStat" scope="LCL" sort="n" key="0"
 scroll="yes" len="*" dtype="CHAR" align="left" skipb="no" />
</cmdrsphdr>
<cmdrspdata>
<rsp>DB(BE3PARTS)MBR(IMSA    ) CC(   0) TYP(DL/I    ) LACC(UPD)
 LSTT(NOTOPEN) </rsp>
<rsp>DB(DEDBJN21)MBR(IMSA    ) CC(   0) TYP(DEDB    ) LACC(UPD)
 LSTT(NOTOPEN) </rsp>
<rsp>DB(DEDBJN21)AREA(DB21AR0 ) MBR(IMSA    ) CC(   0) TYP(AREA)
 LSTT(NOTOPEN) </rsp>
<rsp>DB(DEDBJN21)AREA(DB21AR1 ) MBR(IMSA    ) CC(   0) TYP(AREA)
 LSTT(NOTOPEN) </rsp>
<rsp>DB(DBHDOJ01)MBR(IMSA    ) CC(   0) TYP(PHDAM   ) LACC(UPD) </rsp>
<rsp>DB(DBHDOJ01)PART(PDHDOJA ) MBR(IMSA    ) CC(   0) TYP(PART) LACC(UPD) LSTT(NOTOPEN) </rsp>
<rsp>DB(DBHDOJ01)PART(PDHDOJB ) MBR(IMSA    ) CC(   0) TYP(PART)
 LACC(UPD) LSTT(NOTOPEN) </rsp>
```

```
<rsp>DB(BE3PARTS)MBR(IMS1    ) CC(   0) TYP(DL/I    ) LACC(EXCL)
<rsp>DB(DEDBJN21)MBR(IMS1    ) CC(   0) TYP(DEDB    ) LACC(UPD)
 LSTT(NOTOPEN) </rsp>
<rsp>DB(DEDBJN21)AREA(DB21AR0 ) MBR(IMS1    ) CC(   0) TYP(AREA)
 LSTT(NOTOPEN) </rsp>
<rsp>DB(DEDBJN21)AREA(DB21AR1 ) MBR(IMS1    ) CC(   0) TYP(AREA)
 LSTT(NOTOPEN) </rsp>
<rsp>DB(DBHDOJ01)MBR(IMS1    ) CC(   0) TYP(PHDAM   ) LACC(UPD) </rsp>
<rsp>DB(DBHDOJ01)PART(PDHDOJA ) MBR(IMS1    ) CC(   0) TYP(PART)
 LACC(UPD) LSTT(NOTOPEN) </rsp>
<rsp>DB(DBHDOJ01)PART(PDHDOJB ) MBR(IMS1    ) CC(   0) TYP(PART)
 LACC(UPD) LSTT(NOTOPEN) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: The `QRY DB NAME(BE3PARTS,DEDBJN21,DBHDOJ01) SHOW(ALL)` command returns all the information for the databases BE3PARTS, DEDBJN21 and DBHDOJ01. The database DEDBJN21 is a DEDB and so the DEDB database and all its areas are also returned. The database DBHDOJ01 is a partitioned HDAM database master and so the master database and all its partition are returned.

Example 2 for QUERY DB

TSO SPOC input:

```
QRY DB TYPE(PSINDEX)
```

TSO SPOC output:

```
DBName   PartName MbrName    CC TYPE
D2XHDJ05          IMSA        0 PSINDEX
D2XHDJ05 P2XHDJA  IMSA        0 PART
D2XHDJ05          IMS1        0 PSINDEX
D2XHDJ05 P2XHDJA  IMS1        0 PART
D2XHIDK5          IMSA        0 PSINDEX
D2XHIDK5 P2XIDKA  IMSA        0 PART
D2XHIDK5          IMS1        0 PSINDEX
D2XHIDK5 P2XIDKA  IMS1        0 PART
```

OM API input:

```
CMD(QRY DB TYPE(PSINDEX))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.2.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2003.128 19:36:03.155700</statime>
<stotime>2003.128 19:36:03.157109</stotime>
<staseq>B963C81C46EF4688</staseq>
<stoseq>B963C81C47475E88</stoseq>
<rqsttkn1>USRT005 10123603</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<master>IMSA    </master>
<userid>USRT005 </userid>
<verb>QRY </verb>
<kwd>DB              </kwd>
<input>QRY DB TYPE(PSINDEX) </input>
</cmd>
<cmdrsphdr>
<hdr slbl="DB" llbl="DBName" scope="LCL" sort="a" key="1" scroll="no"
len="8" dtype="CHAR" align="left" skipb="no" />
```

```
<hdr slbl="AREA" llbl="AreaName" scope="LCL" sort="a" key="3"
scroll="no" len="8" dtype="CHAR" align="left" skipb="yes" />
<hdr slbl="PART" llbl="PartName" scope="LCL" sort="a" key="4"
scroll="no" len="8" dtype="CHAR" align="left" skipb="yes" />
<hdr slbl="MBR" llbl="MbrName" scope="LCL" sort="a" key="2" scroll="no"
len="8" dtype="CHAR" align="left" skipb="no" />
<hdr slbl="CC" llbl="CC" scope="LCL" sort="n" key="0" scroll="yes"
len="4" dtype="INT" align="right" skipb="no" />
<hdr slbl="TYP" llbl="TYPE" scope="LCL" sort="n" key="0" scroll="yes"
len="8" dtype="CHAR" align="left" skipb="no" />
</cmdrsphdr>
<cmdrspdata>
<rsp>DB(D2XHDJ05)MBR(IMSA    ) CC(   0) TYP(PSINDEX ) </rsp>
<rsp>DB(D2XHDJ05)PART(P2XHDJA ) MBR(IMSA    ) CC(   0) TYP(PART) </rsp>
<rsp>DB(D2XHIDK5)MBR(IMSA    ) CC(   0) TYP(PSINDEX ) </rsp>
<rsp>DB(D2XHIDK5)PART(P2XIDKA ) MBR(IMSA    ) CC(   0) TYP(PART) </rsp>
<rsp>DB(D2XHDJ05)MBR(IMS1    ) CC(   0) TYP(PSINDEX ) </rsp>
<rsp>DB(D2XHDJ05)PART(P2XHDJA ) MBR(IMS1    ) CC(   0) TYP(PART) </rsp>
<rsp>DB(D2XHIDK5)MBR(IMS1    ) CC(   0) TYP(PSINDEX ) </rsp>
<rsp>DB(D2XHIDK5)PART(P2XIDKA ) MBR(IMS1    ) CC(   0) TYP(PART) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: The `QRY DB TYPE(PSINDEX)` command returns information of all the databases defined as `TYPE(PSINDEX)`. The master database name and all its partitions along with the completion code, the IMS that processed the command, and the database type are returned.

Example 3 of QUERY DB

TSO SPOC input:

```
QRY DB STATUS(STOUPDS)
```

TSO SPOC output:

```
DBName   PartName MbrName    CC TYPE     LclStat
BANKATMS          IMS1        0          NOTINIT,NOTOPEN,STOSCHD,STOUPDS
DBHDOJ01 PDHDOJA  IMSA        0 PART     NOTOPEN,STOUPDS
DBHDOJ01 PDHDOJA  IMS1        0 PART     NOTOPEN,STOUPDS
```

OM API input:

```
CMD(QRY DB STATUS(STOUPDS))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.2.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2003.128 20:00:38.874790</statime>
<stotime>2003.128 20:00:38.876039</stotime>
<staseq>B963CD9BA1EA6F66</staseq>
<stoseq>B963CD9BA238752D</stoseq>
<rqsttkn1>USRT005 10130038</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<master>IMSA    </master>
<userid>USRT005 </userid>
<verb>QRY </verb>
<kwd>DB              </kwd>
<input>QRY DB STATUS(STOUPDS) </input>
</cmd>
<cmdrsphdr>
```

```
<hdr slbl="DB" llbl="DBName" scope="LCL" sort="a" key="1" scroll="no"
 len="8" dtype="CHAR" align="left" skipb="no" />
<hdr slbl="AREA" llbl="AreaName" scope="LCL" sort="a" key="3"
 scroll="no" len="8" dtype="CHAR" align="left" skipb="yes" />
<hdr slbl="PART" llbl="PartName" scope="LCL" sort="a" key="4"
 scroll="no" len="8" dtype="CHAR" align="left" skipb="yes" />
<hdr slbl="MBR" llbl="MbrName" scope="LCL" sort="a" key="2" scroll="no"
 len="8" dtype="CHAR" align="left" skipb="no" />
<hdr slbl="CC" llbl="CC" scope="LCL" sort="n" key="0" scroll="yes"
 len="4" dtype="INT" align="right" skipb="no" />
<hdr slbl="TYP" llbl="TYPE" scope="LCL" sort="n" key="0" scroll="yes"
 len="8" dtype="CHAR" align="left" skipb="no" />
<hdr slbl="LSTT" llbl="LclStat" scope="LCL" sort="n" key="0"
 scroll="yes" len="*" dtype="CHAR" align="left" skipb="no" />
</cmdrsphdr>
<cmdrspdata>
<rsp>DB(DBHDOJ01)PART(PDHDOJA ) MBR(IMSA    ) CC(   0) TYP(PART    )
 LSTT(NOTOPEN,STOUPDS) </rsp>
<rsp>DB(BANKATMS)MBR(IMS1    ) CC(   0) TYP(        )
 LSTT(NOTINIT,NOTOPEN,STOSCHD,STOUPDS) </rsp>
<rsp>DB(DBHDOJ01)PART(PDHDOJA ) MBR(IMS1    ) CC(   0) TYP(PART    )
 LSTT(NOTOPEN,STOUPDS) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: The QRY DB STATUS(STOUPDS) command returns information of all the databases that have the status STOUPDS. The database name, the completion code, the IMS that processed the command, and the database status are returned. Because the database, PDHDOJA, is a partition database that has the STOUPDS status, the HALDB master name is also returned in the DBNAME column.

Example 4 for QUERY DB

TSO SPOC input:

```
QRY DB  NAME(PDHDOJA,PDHDOKA)
```

TSO SPOC output:

```
DBName   PartName MbrName    CC TYPE
         PDHDOJA  IMSA        0 PART
         PDHDOJA  IMS1        0 PART
DBHDOK01 PDHDOKA  IMSA        0 PART
DBHDOK01 PDHDOKA  IMS1        0 PART
```

OM API input:

```
CMD(QRY DB  NAME(PDHDOJA,PDHDOKA))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.2.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2003.128 20:31:10.151252</statime>
<stotime>2003.128 20:31:10.152706</stotime>
<staseq>B963D46E12E54D25</staseq>
<stoseq>B963D46E13402A48</stoseq>
<rqsttkn1>USRT005 10133110</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<master>IMSA    </master>
<userid>USRT005 </userid>
<verb>QRY </verb>
```

```
<kwd>DB              </kwd>
<input>QRY DB  NAME(PDHDOJA,PDHDOKA) </input>
</cmd>
<cmdrsphdr>
<hdr slbl="DB" llbl="DBName" scope="LCL" sort="a" key="1" scroll="no"
 len="8" dtype="CHAR" align="left" skipb="no" />
<hdr slbl="AREA" llbl="AreaName" scope="LCL" sort="a" key="3"
 scroll="no" len="8" dtype="CHAR" align="left" skipb="yes" />
<hdr slbl="PART" llbl="PartName" scope="LCL" sort="a" key="4"
 scroll="no" len="8" dtype="CHAR" align="left" skipb="yes" />
<hdr slbl="MBR" llbl="MbrName" scope="LCL" sort="a" key="2" scroll="no"
 len="8" dtype="CHAR" align="left" skipb="no" />
<hdr slbl="CC" llbl="CC" scope="LCL" sort="n" key="0" scroll="yes"
 len="4" dtype="INT" align="right" skipb="no" />
<hdr slbl="TYP" llbl="TYPE" scope="LCL" sort="n" key="0" scroll="yes"
 len="8" dtype="CHAR" align="left" skipb="no" />
</cmdrsphdr>
<cmdrspdata>
<rsp>DB(        )PART(PDHDOJA ) MBR(IMSA    ) CC(   0) TYP(PART    )
</rsp>
<rsp>DB(DBHDOK01)PART(PDHDOKA ) MBR(IMSA    ) CC(   0) TYP(PART    )
</rsp>
<rsp>DB(        )PART(PDHDOJA ) MBR(IMS1    ) CC(   0) TYP(PART    )
</rsp>
<rsp>DB(DBHDOK01)PART(PDHDOKA ) MBR(IMS1    ) CC(   0) TYP(PART    )
</rsp>
</cmdrspdata>
</imsout>
```

Explanation: The `QUERY DB NAME(PDHDOJA,PDHDOKA)` command is requesting information of the two partition databases, PDHDOKA and PDHDOJA. The partition name is returned in the PartName column and the HALDB master database name is returned in the DBName column (if a master exists). The HALDB master for the database, PDHDOJA, is offline so the database name is returned as blanks. The database type, completion code, and IMS that processed the command is also returned.

QUERY IMSPLEX

`QUERY IMSPLEX` is an IMSplex command that displays information about one or more IMSplex members. If Version 7 IMSs are part of an IMSplex, the `QUERY IMSPLEX` command will not show those IMSs even though they are a part of the IMSplex.

This command can be issued only through the OM API.

NAME()

Specifies the name of the IMSplexes for which member information is to be returned. The IMSplex name may be a generic parameter, to allow easy specification of a group of IMSplexes whose names match a generic parameter mask.

SHOW()

Specifies the output fields to be returned. If SHOW is not specified, only the IMSplex names, IMSplex member names, IMSplex member that builds the output line, and completion codes are returned. This provides a method for a system management application to obtain a list of all IMSplex member names that are currently known in the IMSplexes.

ALL Returns all output fields

JOB Jobname of the IMSplex member

OS Name of the OS image on which the IMSplex member is executing.

STATUS
IMSplex member status.

SUBTYPE
IMSplex member subtype.

TYPE IMSplex member type.

VERSION
IMSplex member version.

STATUS()
Displays IMSplex members that display at least one of the specified status.

ABENDED
IMSplex member has abended

ACTIVE
IMSplex member is active

NOTREACHABLE
The local SCI responsible for the member is not currently active. The status displayed is the current status for the member.

Note: The status output is NOT-REACHABLE (with a hyphen).

OLD The SCI responsible for the member is not currently active. The status displayed is the last known status for the member. The actual status may be different.

READY
IMSplex member is ready to receive messages and requests that are routed by TYPE.

TYPE()
Displays IMSplex members that possess at least one of the specified member types.

AOP Automated Operator Program. Examples of AOPs are a SPOC application that an operator uses to interact with an IMSplex or a program that is monitoring an IMSplex.

BATCH
IMS batch job.

CQS Common Queue Server address space.

DBRC DBRC address space.

IMS IMS region.

IMSCON
An address space that serves as an interface between IMS and a protocol that is not directly supported by IMS (for example, TCP/IP).

OM Operations Manager address space.

OTHER
Other non-IMS address space or job.

RM Resource Manager address space.

SCI Structured Call Interface address space.

QUERY IMSPLEX Output Fields

Table 104 shows the output fields for an QUERY IMSPLEX command. The columns in the table are as follows:

Short Label
Contains the short label generated in the XML output.

Keyword
Identifies the keyword on the command that caused the field to be generated. N/A appears for output fields that are always returned.

Meaning
Provides a brief description of the output field.

Table 104. Output Fields for QUERY IMSPLEX Command

Short Label	Keyword	Meaning
IMSMBR	N/A	IMSplex member name. The IMSplex member name is always returned.
IMSPLX	N/A	IMSplex name. The IMSplex name is always returned.
MBR	N/A	IMSplex member that build the output line. The OM identifier of the OM that built the output line.
CC	N/A	Completion code for the line of output. The completion code is always returned.
STT	STATUS	IMSplex member status. See Table 105 for more information.
JOB	JOB	Jobname of IMSplex member.
OS	OS	Name of OS image on which the IMSplex member is executing.
STYP	SUBTYPE	Subtype of IMSplex member. See Table 107 on page 490 for an explanation of the possible subtypes.
TYP	TYPE	IMSplex member type.
VER	VERSION	IMSplex member version.

QUERY IMSPLEX Status

Table 105 shows the possible IMSplex member status. The table contains information about status such as the STATUS keyword to specify to select members with the specified status, the status that is returned, and the meaning of the status.

Table 105. QUERY IMSPLEX Status Table

Satus Keyword	Status	Meaning
ABENDED	ABENDED	IMSplex member has abended.
ACTIVE	ACTIVE	IMSplex member is active.
NOTREACHABLE	NOT-REACHABLE	The local SCI responsible for the member is currently not active. The status displayed is the current status for the member.
OLD	OLD	The SCI responsible for the member is not currently active. The status displayed is the last known status for the member. The actual status may be different.

Table 105. QUERY IMSPLEX Status Table (continued)

Satus Keyword	Status	Meaning
READY	READY	IMSplex member is ready to receive messages and requests that are routed by TYPE.

QUERY IMSPLEX Types

Table 106 shows the possible IMSplex member types. The table contains information about member types such as the TYPE keyword to specify to select members with the specified type, the type that is returned, and the meaning of the member type.

Table 106. QUERY IMSPLEX Member Types

Type Keyword	Member Type	Meaning
AOP	aop	Automated Operator Program. An example of an AOP is a SPOC (Single Point of Control) that an operator uses to interact with the IMSplex. Another example of an AOP is a program that is monitoring the IMSplex.
BATCH	batch	IMS batch job.
CQS	cqs	Common Queue Server address space. CQS manages shared queues and may also manage resources on a resource structure.
DBRC	dbrc	DBRC address space.
IMS	ims	IMS region.
IMSCON	imscon	IMS connect. An address space that serves as an interface between IMS and a protocol that is not directly supported by IMS.
OM	om	Operations Manager address space. Operations Manager supports IMS operations in an IMSplex.
OTHER	other	Other non-IMS address space or job.
RM	rm	Resource Manager address space. Resource manager supports global resources in an IMSplex.
SCI	sci	Structured Call Interface address space.

QUERY IMSPLEX Subtypes

Table 107 shows the possible IMSplex member subtypes. The table contains information about member types, the member subtypes associated with them, and the meaning of the member subtype. Members omitted from this table do not define a member subtype.

Table 107. QUERY IMSPLEX Member Subtypes

Member Type	Member Subtype	Meaning
IMS	DBDC	IMS DB/DC address space.
IMS	DBCTL	IMS DBCTL address space. DBCTL supports database functions.
IMS	DCCTL	IMS DCCTL address space. DCCTL supports data communications functions.

Table 107. QUERY IMSPLEX Member Subtypes (continued)

Member Type	Member Subtype	Meaning
IMS	FDBR	IMS Fast Database Recovery. An IMS control region that recovers database resources when an IMS database manager fails.
RM	SNGLRM	RM is defined without a resource structure, so only a single RM is allowed in the IMSplex.
RM	MULTRM	RM is defined with a resource structure, so multiple RMs are allowed in the IMSplex.

Return, Reason, and Completion Codes for QUERY IMSPLEX

The return and reason codes that can be returned as a result of the QUERY IMSPLEX command are standard for all commands entered through the OM API. Refer to the OM Return and Reason code section in the *IMS Version 9: Common Service Layer Guide and Reference* for the list of codes and their meanings.

Table 108 contains the return, reason, and completion codes for the QUERY IMSPLEX command. Included in the tables is a brief explanation of the codes.

Table 108. Return and Reason Codes for the QUERY IMSPLEX Command

Return Code	Reason Code	Meaning
X'00000000'	X'00000000'	The QUERY IMSPLEX command completed successfully.
X'02000008'	X'00002048'	The QUERY IMSPLEX command has more than one filter value specified. Only one of the TYPE or STATUS filters can be specified.
X'02000008'	X'0000203C'	An invalid keyword parameter value was specified.
X'0200000C'	X'00003000'	The QUERY IMSPLEX command is successful for at least one resource name. The QUERY IMSPLEX command is not successful for one or more resource names. The completion code indicates the reason for the error with the resource name. The completion codes that can be returned by the QUERY IMSPLEX command are listed in the QUERY IMSPLEX completion code table.
X'0200000C'	X'00003004'	No resources were found to be returned. The resource name(s) specified may be invalid or there were no resources that match the filter specified.
X'02000014'	X'00005020'	The QUERY IMSPLEX command processing terminated. OM was unable to obtain storage for a system AWE while processing the command.
X'02000014'	X'0000502C'	The QUERY IMSPLEX command processing terminated. OM was unable to obtain storage for the command output header.
X'02000014'	X'00005030'	The QUERY IMSPLEX command processing terminated. OM was unable to obtain storage for the command output response.
X'02000014'	X'00005040'	The QUERY IMSPLEX command processing terminated due to an SCI error.

Errors unique to the processing of this command are returned as a completion codes. A completion code is returned for each action against an individual resource.

Table 109 contains completion codes can be returned on a QUERY IMSPLEX command.

Table 109. Completion Codes for the QUERY IMSPLEX Command

Completion Code	Meaning
0	The QUERY IMSPLEX command completed successfully for the resource.

Table 109. Completion Codes for the QUERY IMSPLEX Command (continued)

Completion Code	Meaning
4	The IMSplex name is unknown to the client that is processing the request. The imsplex name may have been typed in error or the imsplex may not be active at this time. If this is a wildcard request there were no matches for the name. Confirm the correct spelling of the resource name is specified on the command.

Examples for QUERY IMSPLEX Command

This section provides OM API and TSO SPOC input and output examples and explanations for the QUERY command.

Example 1 for QUERY IMSPLEX Command

TSO SPOC input:

```
QRY IMSPLEX NAME(CSLPLEX1) SHOW(JOB,SUBTYPE,STATUS,TYPE)
```

TSO SPOC output:

```
Response for: QUERY IMSPLEX NAME(CSLPLEX1) SHOW(JOB,SUBTYPE,STATUS)
IMSplex  MbrName    CC Member   JobName  Type  Subtype  Status
CSLPLEX1 OM1OM       0 IMS2     IMS2     IMS   DBDC     READY,ACTIVE
CSLPLEX1 OM1OM       0 CQS1CQS  CQSRE1   CQS            ACTIVE
CSLPLEX1 OM1OM       0 SYS3     IMS1     IMS   DBDC     READY,ACTIVE
CSLPLEX1 OM1OM       0 OM1OM    OM1      OM             READY,ACTIVE
CSLPLEX1 OM1OM       0 IMS3     IMS3     IMS   DBDC     READY,ACTIVE
CSLPLEX1 OM1OM       0 USRT011  USRT011  AOP            ACTIVE
CSLPLEX1 OM1OM       0 RM1RM    RM1      RM    MULTRM   READY,ACTIVE
CSLPLEX1 OM1OM       0 SCI1SC   SCI1     SCI            READY,ACTIVE
```

OM API input:

```
CMD(QRY IMSPLEX NAME(CSLPLEX1) SHOW(JOB,SUBTYPE,STATUS,TYPE))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2002.163 15:05:18.859217</statime>
<stotime>2002.163 15:05:18.860443</stotime>
<staseq>B7C4A41E663D11C3</staseq>
<stoseq>B7C4A41E6689B9C3</stoseq>
<rqsttkn1>USRT011 10080518</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<userid>USRT011 </userid>
<verb>QRY </verb>
<kwd>IMSPLEX         </kwd>
<input>QUERY IMSPLEX NAME(CSLPLEX1) SHOW(JOB,SUBTYPE,STATUS,TYPE)</input>
</cmd>
<cmdrsphdr>
<hdr slbl="IMSPLX" llbl="IMSplex" scope="LCL" sort="A" key="1" scroll="NO"  len="8"
dtype="CHAR" align="left"  />
<hdr slbl="MBR"    llbl="MbrName" scope="LCL" sort="N" key="0" scroll="YES" len="8"
dtype="CHAR" align="left"  />
<hdr slbl="CC"     llbl="CC"      scope="LCL" sort="N" key="0" scroll="YES" len="4"
dtype="INT"  align="right" />
<hdr slbl="IMSMBR" llbl="Member"  scope="LCL" sort="N" key="0" scroll="NO"  len="8"
dtype="CHAR" align="left"  />
```

```
<hdr slbl="JOB"    llbl="JobName" scope="LCL" sort="N" key="0" scroll="YES" len="8"
dtype="CHAR" align="left"  />
<hdr slbl="TYP"    llbl="Type"    scope="LCL" sort="N" key="0" scroll="YES" len="5"
dtype="CHAR" align="left"  />
<hdr slbl="STYP"   llbl="Subtype" scope="LCL" sort="N" key="0" scroll="YES" len="8"
dtype="CHAR" align="left"  />
<hdr slbl="STT"    llbl="Status"  scope="GBL" sort="N" key="0" scroll="YES" len="*"
dtype="CHAR" align="left"  />
</cmdrsphdr>
<cmdrspdata>
<rsp>IMSPLX(CSLPLEX1) MBR(OM1OM)  IMSMBR(IMS2)    CC(   0) JOB(IMS2)    TYP(IMS)
STYP(DBDC) STT(READY,ACTIVE)</rsp>
<rsp>IMSPLX(CSLPLEX1) MBR(OM1OM)  IMSMBR(CQS1CQS) CC(   0) JOB(CQSRE1)  TYP(CQS)
STYP(  ) STT(ACTIVE)</rsp>
<rsp>IMSPLX(CSLPLEX1) MBR(OM1OM)  IMSMBR(SYS3)    CC(   0) JOB(IMS1)    TYP(IMS)
STYP(DBDC) STT(READY,ACTIVE)</rsp>
<rsp>IMSPLX(CSLPLEX1) MBR(OM1OM)  IMSMBR(OM1OM)   CC(   0) JOB(OM1)     TYP(OM)
STYP(  ) STT(EADY,ACTIVE)</rsp>
<rsp>IMSPLX(CSLPLEX1) MBR(OM1OM)  IMSMBR(IMS3)    CC(   0) JOB(IMS3)    TYP(IMS)
STYP(DBDC) STT(READY,ACTIVE)</rsp>
<rsp>IMSPLX(CSLPLEX1) MBR(OM1OM)  IMSMBR(USRT011) CC(   0) JOB(USRT011) TYP(AOP)
STYP(  ) STT(ACTIVE)</rsp>
<rsp>IMSPLX(CSLPLEX1) MBR(OM1OM)  IMSMBR(RM1RM)   CC(   0) JOB(RM1)     TYP(RM)
STYP(MULTRM) STT(READY,ACTIVE)</rsp>
<rsp>IMSPLX(CSLPLEX1) MBR(OM1OM)  IMSMBR(SCI1SC)  CC(   0) JOB(SCI1)    TYP(SCI)
STYP(  ) STT(READY,ACTIVE)</rsp>
</cmdrspdata>
</imsout>
```

Explanation: The QUERY IMSPLEX command displays the IMSplex members that compose IMSplex CSLPLEX1. This IMSplex contains three IMSs (IMS1, IMS2, and IMS3), a TSO SPOC (USRT011), a CQS (CQSRE1), RM (RM1), and OM (OM1). OM1 is the command master that built the output.

Example 2 for QUERY IMSPLEX Command

TSO SPOC input:

```
QRY IMSPLEX NAME(CSLPLEX1) SHOW(JOB,SUBTYPE,STATUS,TYPE)
```

TSO SPOC output:

```
 Response for: QUERY IMSPLEX NAME(CSLPLEX1) SHOW(JOB,SUBTYPE,STATUS,TYPE)
IMSplex  MbrName    CC Member   JobName  Type  Subtype  Status
CSLPLEX1 OM1OM       0 IMS2     IMS2     IMS   DBDC     READY,ACTIVE
CSLPLEX1 OM1OM       0 CQS1CQS  CQSRE1   CQS            ACTIVE
CSLPLEX1 OM1OM       0 SYS3     IMS1     IMS   DBDC     READY,ACTIVE
CSLPLEX1 OM1OM       0 OM1OM    OM1      OM             READY,ACTIVE
CSLPLEX1 OM1OM       0 IMS3     IMS3     IMS   DBDC     READY,ACTIVE
CSLPLEX1 OM1OM       0 USRT011  USRT011  AOP            ACTIVE
CSLPLEX1 OM1OM       0 RM1RM    RM1      RM    MULTRM   READY,ACTIVE
CSLPLEX1 OM1OM       0 SCI1SC   SCI1     SCI            READY,ACTIVE
```

OM API input:

```
CMD (QRY IMSPLEX NAME(CSLPLEX1) SHOW(JOB,SUBTYPE,STATUS,TYPE))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname><omvsn>1.1.0</omvsn>
<xmlvsn>1</xmlvsn>
<statime>2002.163 15:05:18.859217</statime>
<stotime>2002.163 15:05:18.860443</stotime>
<staseq>B7C4A41E663D11C3</staseq>
<stoseq>B7C4A41E6689B9C3</stoseq>
<rqsttkn1>USRT011 10080518</rqsttkn1>
```

```
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<userid>USRT011 </userid>
<verb>QRY </verb>
<kwd>IMSPLEX         </kwd>
<input>QUERY IMSPLEX NAME(CSLPLEX1) SHOW(JOB,SUBTYPE,STATUS,TYPE)</input>
</cmd>
<cmdrsphdr>
<hdr slbl="IMSPLX" llbl="IMSplex" scope="LCL" sort="A" key="1" scroll="NO"  len="8"
dtype="CHAR" align="left" />
<hdr slbl="MBR"    llbl="MbrName" scope="LCL" sort="N" key="0" scroll="YES" len="8"
dtype="CHAR" align="left" />
<hdr slbl="CC"     llbl="CC"      scope="LCL" sort="N" key="0" scroll="YES" len="4"
dtype="INT"  align="right" />
<hdr slbl="IMSMBR" llbl="Member"  scope="LCL" sort="N" key="0" scroll="NO"  len="8"
dtype="CHAR" align="left" />
<hdr slbl="JOB"    llbl="JobName" scope="LCL" sort="N" key="0" scroll="YES" len="8"
dtype="CHAR" align="left" />
<hdr slbl="TYP"    llbl="Type"    scope="LCL" sort="N" key="0" scroll="YES" len="5"
dtype="CHAR" align="left" />
<hdr slbl="STYP"   llbl="Subtype" scope="LCL" sort="N" key="0" scroll="YES" len="8"
dtype="CHAR" align="left" />
<hdr slbl="STT"    llbl="Status"  scope="GBL" sort="N" key="0" scroll="YES" len="*"
dtype="CHAR" align="left" />
</cmdrsphdr>
<cmdrspdata>
<rsp>IMSPLX(CSLPLEX1) MBR(OM1OM) IMSMBR(IMS2)    CC(   0) JOB(IMS2)   TYP(IMS)
STYP(DBDC) STT(READY,ACTIVE)</rsp>
<rsp>IMSPLX(CSLPLEX1) MBR(OM1OM) IMSMBR(CQS1CQS) CC(   0) JOB(CQSRE1) TYP(CQS)
STYP(   ) STT(ACTIVE)</rsp>
<rsp>IMSPLX(CSLPLEX1) MBR(OM1OM) IMSMBR(SYS3)    CC(   0) JOB(IMS1)   TYP(IMS)
STYP(DBDC) STT(READY,ACTIVE)</rsp>
<rsp>IMSPLX(CSLPLEX1) MBR(OM1OM) IMSMBR(OM1OM)   CC(   0) JOB(OM1)    TYP(OM)
STYP(   ) STT(READY,ACTIVE)</rsp>
<rsp>IMSPLX(CSLPLEX1) MBR(OM1OM) IMSMBR(IMS3)    CC(   0) JOB(IMS3)   TYP(IMS)
STYP(DBDC) STT(READY,ACTIVE)</rsp>
<rsp>IMSPLX(CSLPLEX1) MBR(OM1OM) IMSMBR(USRT011) CC(   0) JOB(USRT011) TYP(AOP)
STYP(    ) STT(ACTIVE)</rsp>
<rsp>IMSPLX(CSLPLEX1) MBR(OM1OM) IMSMBR(RM1RM)   CC(   0) JOB(RM1)    TYP(RM)
STYP(MULTRM) STT(READY,ACTIVE)</rsp>
<rsp>IMSPLX(CSLPLEX1) MBR(OM1OM) IMSMBR(SCI1SC)  CC(   0) JOB(SCI1)   TYP(SCI)
STYP(    ) STT(READY,ACTIVE)</rsp>
</cmdrspdata>
</imsout>
```

Explanation: The QUERY IMSPLEX command displays the IMSplex members that compose IMSplex CSLPLEX1. This IMSplex contains three IMSs (IMS1, IMS2, and IMS3), a TSO SPOC (USRT011), a CQS (CQSRE1), RM (RM1), and OM (OM1). OM1 is the command master that built the output.

QUERY LE

The QUERY LE command allows the user to display Language Environment (LE) runtime parameter overrides defined by a previous UPDATE LE command. The query can use filters on transaction code, LTERM, userid, or program name. Any combination of parameters may be used to qualify the application instance. All entries found that match the criteria are returned to the user. The user specifies on the command which output fields should be returned in the command response. The user can ask for all information that includes transaction code, LTERM name, userid, program name, and runtime parameters. See Chapter 65, “UPDATE,” on page 705 for more information about the UPDATE LE command.

This command may be specified only through the Operations Manager API.

The command syntax for this command is defined in XML and is available to automation programs that communicate with OM.

The following parameters support a generic or wildcard parameter. A generic parameter is a 1-8 character name that includes an asterisk or a percent sign. An asterisk can be replaced by a zero or more characters to create a valid resource name. A percent sign can be replaced by exactly one character to create a valid resource name.

LTERM()
Specifies the 1-8 character name of the LTERM or LTERMs matching the generic or wildcard parameter.

PGM() Specifies the 1-8 character name of the program or programs matching the generic or wildcard parameter.

SHOW()
Specifies the output fields to be returned. At least one SHOW field is required on the command.

ALL Returns all the output fields. This is the same as if the following was specified: `SHOW(TRAN,LTERM,USERID,PGM,LERUNOPTS)`.

LERUNOPTS
Returns all of the LE override parameters associated with the transaction, LTERM, userid, or program name.

LTERM
Returns the logical terminal name field.

PGM Returns the program name field.

TRAN Returns the transaction name field.

USERID
Returns the user identifier field.

TRAN()
Specifies the 1-8 character name of the transaction or transactions matching the generic or wildcard parameter. If the TRAN, LTERM, USERID, or PGM resource filters are not specified, all parameter overrides are returned.

USERID()
Specifies the 1-8 character name of the userid or userids matching the generic or wildcard parameter. If the TRAN, LTERM, USERID, or PGM resource filters are not specified, all parameter overrides are returned.

QUERY LE Output Fields

Table 110 on page 497 shows the `QUERY LE` output fields. The columns in the table are as follows:

Short Label
Contains the short label generated in the XML output.

Keyword
Identifies the keyword on the command that caused the field to be generated. N/A appears for output fields that are always returned.

Meaning

Provides a brief description of the output field.

Table 110. Output Fields for QUERY LE Command

Short Label	Keyword	Meaning
CC	N/A	Completion code for the line of output. Completion code is always returned.
LTRM	LTERM	LTERM Name requested by the QUERY.
MBR	N/A	IMSPlex member (IMS identifier) that built the output line. Member name is always returned.
PGM	PGM	Program Name requested by the QUERY.
PRM	LERUNOPTS	The LE override parameters for the specified resource filters.
TRAN	TRAN	Transaction Name requested by the QUERY.
UID	USERID	Userid requested by the QUERY.

Return, Reason, and Completion Codes for QUERY LE

An IMS return and reason code is returned to OM by the QUERY LE command. The OM return and reason codes that may be returned as a result of the QUERY LE command are standard for all commands entered through the OM API. Table 111 includes the return and reason codes and a brief explanation of the codes. Table 112 includes an explanation of the completion code.

Table 111. Return and Reason Codes for the QUERY LE command

Return Code	Reason Code	Meaning
X'00000000'	X'00000000'	The QUERY LE command completed successfully.
X'00000008'	X'0000200C'	No resources found to return. Either the entry was previously deleted or a keyword filter was typed incorrectly.
X'00000008'	X'00002014'	An invalid character was specified in the filter name.
X'00000010'	X'00004040'	The parameter override header has not been initialized. Retry the command after restart is complete.
X'00000014'	X'00005000'	Unable to get storage from IMODULE GETSTOR.
X'00000014'	X'00005010'	Unable to obtain latch.
X'00000014'	X'00005FFF'	Internal IMS Error - Should not occur.

Table 112. Completion code for the QUERY LE command

Completion code	Meaning
0	The QUERY LE command completed successfully for the specified resource.

Examples for QUERY LE Command

This section provides OM API and TSO SPOC input and output examples and explanations for the QUERY command.

Example 1 for QUERY LE Command

Assume the following filters are specified on QRY LE commands:

1. TRAN(PART) SHOW(ALL) Returns entries #1, 2, 3, 5, 6, 8.
2. TRAN(PART) LTERM(TERM1) SHOW(ALL) Returns entries #3, 5, 6.
3. LTERM(TERM2) USERID(BETTY) SHOW(ALL) Returns entry #7.
4. TRAN(PART) LTERM(TERM1) USERID(BETTY) SHOW(ALL) Does not return any entries.
5. TRAN(PART) LTERM(TERM*) SHOW(ALL) Returns entries #3, 5, 6, 8.
6. USERID(B*) SHOW(ALL) Returns entries #2, 5, 6, 7.

Rules for matching an entry which results in it being returned on QUERY command:

- If a filter is specified on the command for a particular resource it must match the resource filter defined in the entry. The resource in the QUERY LE command may be specified with wildcards as defined previously.
- A resource filter that is not specified on a QUERY LE command will match on any filter for the specific resource defined in the entry. A non-specified filter is treated as a wildcard. For instance if the LTERM filter is not specified on a QRY LE command it will match on any LTERM resource defined in an entry, as if LTERM(*) was specified on the command.

Table 113 is a logical representation of the parameter override table entries prior to any of the above query commands being processed.

Table 113. Parameter Override Table Entries for Example 1

Entry#	TRAN	LTERM	USERID	PROGRAM	LERUNOPTS
1	PART			DFSSAM02	aaaa
2	PART		BETTY		bbbb
3	PART	TERM1			cccc
4				DFSSAM02	dddd
5	PART	TERM1	BARBARA		eeee
6	PART	TERM1	BOB		ffff
7		TERM2	BETTY		gggg
8	PART	TERM2			iiii

Example 2 for QUERY LE Command

TSO SPOC input:

```
QRY LE SHOW(ALL)
```

TSO SPOC output:

```
SYS3 0                          IAPMDI29 CCCC
SYS3 0 IAPMDI26      USRT001      RPTOPTS=((ON),NOOVR),RPTSTG=(((OFF),NOOVR)
SYS3 0 IAPMDI27 IMS1 USRT001  IAPMDI27 AAAA
```

OM API input:

```
CMD(QRY LE SHOW(ALL))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
```

```
<xmlvsn>1   </xmlvsn>
<statime>2002.163 17:34:01.196902</statime>
<stotime>2002.163 17:34:01.197368</stotime>
<staseq>B7C4C55B67566505</staseq>
<stoseq>B7C4C55B67738365</stoseq>
<rqsttkn1>USRT002 10103401</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<master>SYS3    </master>
<userid>USRT002 </userid>
<verb>QRY </verb>
<kwd>LE              </kwd>
<input>QRY LE SHOW(ALL)          </input>
</cmd>
<cmdrsphdr>
<hdr slbl="MBR"  llbl="MbrName"   scope="LCL" sort="A" key="1" scroll="NO" len="8"
dtype="CHAR" align="left" />
<hdr slbl="CC"   llbl="CC"        scope="LCL" sort="N" key="0" scroll="YES" len="4"
dtype="INT" align="right" />
<hdr slbl="TRAN" llbl="Trancode"  scope="LCL" sort="N" key="0" scroll="YES" len="8"
dtype="CHAR" align="left" />
<hdr slbl="LTRM" llbl="Lterm"     scope="LCL" sort="N" key="0" scroll="YES" len="8"
dtype="CHAR " align="left" />
<hdr slbl="UID"  llbl="Userid"    scope="LCL" sort="N" key="0" scroll="YES" len="8"
dtype="CHAR " align="left" />
<hdr slbl="PGM"  llbl="Program"   scope="LCL" sort="N" key="0" scroll="YES" len="8"
dtype="CHAR" align="left" />
<hdr slbl="PRM"  llbl="LERunOpts" scope="LCL" sort="N" key="0" scroll="YES" len="*"
dtype="CHAR" align="left" />
</cmdrsphdr>
<cmdrspdata>
<rsp>MBR(SYS3    ) CC(   0) TRAN(        ) LTRM(      ) UID(        ) PGM(IAPMDI29)
PRM(CCCC    ) </rsp>
<rsp>MBR(SYS3    ) CC(   0) TRAN(IAPMDI26) LTRM(      ) UID(USRT001 ) PGM(        )
PRM(RPTOPTS=((ON),NOOVR),RPTSTG=((OFF),NOOVR)       )</rsp>
<rsp>MBR(SYS3    ) CC(   0) TRAN(IAPMDI27) LTRM(IMS1  ) UID(USRT001 ) PGM(IAPMDI27)
PRM(AAAA    ) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: The SHOW(ALL) parameter is specified, so all four filters and the runtime option string are shown for each table entry. Furthermore, no filters are specified in the command, so all table entries are shown. In this example, there are three table entries. The first specifies one filter (program) and the parameter string for this entry is CCCC. The second entry specifies two filters, trancode and userid, and its parameter string is RPTOPTS=((ON),NOOVR),RPTSTG=((OFF),NOOVR). The last entry specifies all four filters and a parameter string of AAAA.

Example 3 for QUERY LE Command

TSO SPOC input:

```
QRY LE SHOW(LTERM,USERID)
```

TSO SPOC output:

```
MbrName    CC Lterm    Userid
SYS3        0
SYS3        0          USRT001
SYS3        0 IMS1     USRT001
```

OM API input:

```
CMD(QRY LE SHOW(LTERM,USERID))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2002.163 17:36:27.588393</statime>
<stotime>2002.163 17:36:27.589261</stotime>
<staseq>B7C4C5E703729D6F</staseq>
<stoseq>B7C4C5E703A8D467</stoseq>
<rqsttkn1>USRT002 10103627</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<master>SYS3    </master>
<userid>USRT002 </userid>
<verb>QRY </verb>
<kwd>LE              </kwd>
<input>QRY LE SHOW(LTERM,USERID)          </input>
</cmd>
<cmdrsphdr>
<hdr slbl="MBR"  llbl="MbrName" scope="LCL" sort="A" key="1" scroll="NO"  len="8"
dtype="CHAR" align="left" />
<hdr slbl="CC"   llbl="CC"      scope="LCL" sort="N" key="0" scroll="YES" len="4"
dtype="INT" align="right" />
<hdr slbl="LTRM" llbl="Lterm"   scope="LCL" sort="N" key="0" scroll="YES" len="8"
dtype="CHAR " align="left" />
<hdr slbl="UID"  llbl="Userid"  scope="LCL" sort="N" key="0" scroll="YES" len="8"
dtype="CHAR " align="left" />
</cmdrsphdr>
<cmdrspdata>
<rsp>MBR(SYS3    ) CC(   0) LTRM(        ) UID(        ) </rsp>
<rsp>MBR(SYS3    ) CC(   0) LTRM(        ) UID(USRT001 ) </rsp>
<rsp>MBR(SYS3    ) CC(   0) LTRM(IMS1    ) UID(USRT001 ) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: This command uses the SHOW parameter to limit the amount of data that is shown for each entry in the table. All three table entries are shown, but only the LTERM and TRAN filters are shown for each one. The first entry has neither an LTERM filter nor a USERID filter defined, so it is blank except for the MbrName and CC.

Example 4 for QUERY LE Command

TSO SPOC input:

```
QRY LE USERID(USRT*) SHOW(LTERM,USERID)
```

TSO SPOC output:

```
MbrName    CC Lterm    Userid
SYS3        0          USRT001
SYS3        0 IMS1     USRT001
```

OM API input:

```
CMD(QRY LE USERID(USRT*) SHOW(LTERM,USERID))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1   </xmlvsn>
```

```
<statime>2002.163 17:50:24.925819</statime>
<stotime>2002.163 17:50:24.926381</stotime>
<staseq>B7C4C9058F87B484</staseq>
<stoseq>B7C4C9058FAAD324</stoseq>
<rqsttkn1>USRT002 10105024</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<master>SYS3    </master>
<userid>USRT002 </userid>
<verb>QRY </verb>
<kwd>LE   </kwd>
<input>QRY LE USERID(USRT*) SHOW(LTERM,USERID)  </input>
</cmd>
<cmdrsphdr>
<hdr slbl="MBR"  llbl="MbrName" scope="LCL" sort="A" key="1" scroll="NO"  len="8"
dtype="CHAR" align="left" />
<hdr slbl="CC"   llbl="CC"      scope="LCL" sort="N" key="0" scroll="YES" len="4"
dtype="INT" align="right" />
<hdr slbl="LTRM" llbl="Lterm"   scope="LCL" sort="N" key="0" scroll="YES" len="8"
dtype="CHAR " align="left" />
<hdr slbl="UID"  llbl="Userid"  scope="LCL" sort="N" key="0" scroll="YES" len="8"
dtype="CHAR " align="left" />
</cmdrsphdr>
<cmdrspdata>
<rsp>MBR(SYS3    ) CC(   0) LTRM(        ) UID(USRT001 ) </rsp>
<rsp>MBR(SYS3    ) CC(   0) LTRM(IMS1    ) UID(USRT001 ) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: This command uses both the USERID filter and the SHOW parameter to limit the amount of data shown in the output. The USERID(USRT*) parameter limits the output to only those table entries that define a USERID filter that fits the specified pattern (USRT*). The SHOW parameter limits how much information is returned about each table entry In this case, only the LTERM and USERID are shown.

QUERY MEMBER

The QUERY MEMBER command displays status or attribute information about one or more members of the IMSplex. For IMS Version 9, TYPE(IMS) is the only type supported. QUERY MEMBER can be specified only through the OM API.

SHOW()

Specifies the output fields to be returned.

ALL Returns all the output fields.

ATTRIB

Displays the IMSplex member attributes. These are static definitions.

MODID

Displays the online change modify id. The modify id is incremented by each successful online change. During a global online change, an IMS's local modify id indicates whether the IMS has reached the online change commit phase 2 and is synchronized with the OLCSTAT data set. If the modify id is initialized to zero by the Global Online Change utility, the modify id represents the number of global online changes that have successfully completed.

The local modify id on an XRF alternate system indicates whether the XRF alternate has gotten the X'70' log record, performed online change, and is synchronized with the OLCSTAT data set. If the XRF alternate's modid matches the OLCSTAT data set modid as displayed by the QUERY OLC LIBRARY (OLCSTAT) command, the XRF alternate is synchronized with the OLCSTAT data set.

STATUS
Displays the IMSplex member status. IMSplex member status can change dynamically.

TYPE Shows the IMSplex member type. The IMSplex member type can be IMS, for the IMS address space.

TYPE()
Specifies the IMSplex member type for which information will be displayed.

QUERY MEMBER Output Fields

Table 114 shows information about the QUERY MEMBER TYPE(IMS) output fields. The columns in the table are as follows:

Short Label
Contains the short label generated in the XML output.

Keyword
Identifies the keyword on the command that caused the field to be generated. N/A appears for output fields that are always returned.

Scope Identifies the scope of the output field.

Meaning
Provides a brief description of the output field.

Table 114. Output Fields for QUERY MEMBER Command

Short Label	Keyword	Scope	Meaning
CC	N/A	N/A	Completion code for the line of output. The completion code is always returned.
MBR	N/A	N/A	IMSplex member that built the output line. IMS identifier of IMS that built the output. The IMS identifier is always returned.
LATTR	ATTRIB	LCL	Local IMS attributes. See "QUERY MEMBER Attributes" on page 503 for more information.
MODI	MODID	LCL	Online change modify id, which is incremented by 1 for each online change.
LSTT	STATUS	LCL	Local IMS status. See "QUERY MEMBER Status" on page 503 for more information.
STT	STATUS	GBL	Global IMS status. See "QUERY MEMBER Status" on page 503 for more information.
TYP	TYPE	LCL	IMSplex member type. The IMSplex member type can be IMS, for the IMS address space.

QUERY MEMBER Attributes

Table 115 shows the possible IMS attributes. The table contains information about attributes such as the attribute that is returned, the scope of the attribute, and the meaning of the attribute. Global attributes are returned with the ATTR short label. Local attributes are returned with the LATTR short label.

Table 115. Attributes for QUERY MEMBER Command

Attribute	Scope	Meaning
GBLOLC	LCL	Global online change is enabled.
RMENVNO	LCL	No RM environment is running.
RSRTRK	LCL	Remote Site Recovery tracker.
SHAREDQ	LCL	Shared queues are enabled.

QUERY MEMBER Status

Table 116 shows the possible member status. The table contains information about status such as the status that is returned, the scope of the status, and the meaning of the status. Global status is returned with the STT short label. Local status is returned with the LSTT short label. A scope of LCL means that the status is local to the IMS specified and is returned with the LSTT short label. A scope of GBL means that the status is global to all the IMSs and is returned with the STT short label.

Table 116. Status for QUERY MEMBER Command

Status	Scope	Meaning
LEOPT	LCL	Language Environment options are enabled for this IMS.
OLCABRTC	LCL	Online change abort completed. A `TERMINATE OLC` command or `/MODIFY ABORT` command is entered. Online change abort phase completed locally for this IMS. The IMS is taken out of the online change state.
OLCABRTI	LCL	Online change abort in progress. A `TERMINATE OLC` command or `/MODIFY ABORT` command is entered. Online change abort phase is in progress locally for this IMS.
OLCCMT1C	LCL or GBL	Online change commit phase 1 completed. An `INITIATE OLC PHASE(COMMIT)` command or `/MODIFY COMMIT` command is entered. Online change commit phase 1 completed either locally for the IMS, or globally for all of the IMSs in the IMSplex. After all of the IMSs have attempted commit phase 1, the online change master updates the OLCSTAT data set and the online change is considered to be complete. Classic commands from the system console, an IMS terminal, or the MTO are queued while the IMS is in this state. Queued commands are processed after the online change is committed or aborted. If the classic command is entered from the system console, the WTOR does not appear until this IMS is out of the online change state.

Table 116. Status for QUERY MEMBER Command (continued)

Status	Scope	Meaning
OLCCMT1I	LCL or GBL	Online change commit phase 1 in progress. An INITIATE OLC PHASE(COMMIT) command or /MODIFY COMMIT command is entered. Online change commit phase 1 is in progress either locally for this IMS or globally for all the IMSs in the IMSplex.
OLCCMT2C	LCL or GBL	Online change commit phase 2 completed. An INITIATE OLC PHASE(COMMIT) command or a /MODIFY COMMIT command is entered. Online change commit phase completed either locally for this IMS or globally for all the IMSs in the IMSplex. Classic commands from the system console, an IMS terminal, or the MTO are queued while the IMS is in this state. Queued commands are processed after the online change is committed or aborted. If the classic command is entered from the system console, the WTOR does not appear until this IMS is out of the online change state.
OLCCMT2F	LCL	Online change commit phase 2 failed. An INITIATE OLC PHASE(COMMIT) command or a /MODIFY COMMIT command is entered. Online change commit phase 2 failed locally for this IMS. This IMS may be stuck in an online change state, where the TERMINATE OLC command or /MODIFY ABORT command does not work. If that is the case, cancel the IMS and warm start IMS. This IMS can warm start, since it successfully participated in the online change except for commit phase 2.
OLCCMT2I	LCL or GBL	Online change commit phase 2 in progress. An INITIATE OLC PHASE(COMMIT) command or a /MODIFY COMMIT command is entered. Online change commit phase 2 is in progress either locally for this IMS or globally for all the IMSs in the IMSplex.
OLCMSTR	GBL	Online change phase master. An INITIATE OLC PHASE(PREPARE), an INITIATE OLC PHASE(COMMIT), or a TERMINATE OLC command is entered. This IMS is the master of the online change phase currently in progress, either prepare, commit, or terminate. A different IMS may be master of each phase of online change.
OLCPREPC	LCL or GBL	Online change prepare phase completed. An INITIATE OLC PHASE(PREPARE) command or a /MODIFY PREPARE command is entered. Online change prepare phase completed locally for this IMS or globally for all the IMSs in the IMSplex.
OLCPREPF	LCL	Online change prepare phase failed. An INITIATE OLC PHASE(PREPARE) command is entered. Online change prepare phase failed locally for this IMS. A TERMINATE OLC is requlred to delete the MWA created for the online change and also to delete the online change process that was initiated with RM.

Table 116. Status for QUERY MEMBER Command (continued)

Status	Scope	Meaning
OLCPREPI	LCL or GBL	Online change prepare phase in progress. An INITIATE OLC PHASE(PREPARE) command or a /MODIFY PREPARE command is entered. Online change prepare phase is in progress locally for this IMS or globally for all the IMSs in the IMSplex.
OLCTERMC	GBL	Online change terminate completed. A TERMINATE OLC command was entered. Online change termination is completed for the IMSs in the IMSplex.
OLCTERMF	LCL	TERMINATE FAILED: A TERMINATE OLC command is directed to the IMS that is not in an online change state. An MWA is created to coordinate the TERMINATE OLC command. The TERMINATE OLC command fails due to an RM, SCI, or CQS error and the MWA is set to a 'Terminate Failed' state. A subsequent TERMINATE OLC command is required to delete the MWA in this state.
OLCTERMI	GBL	Online change terminate in progress. A TERMINATE OLC command is entered. Online change termination is in progress for the IMSs in the IMSplex. Online change termination aborts the online change. If all of the IMSs are in an online change prepare state, TERMINATE OLC aborts the online change and removes all of the IMSs from the online change state. If an error occurs before the OLCSTAT data set is updated, then TERMINATE OLC aborts the online change. The online change abort phase is performed on the IMSs where abort is needed. All of the IMSs are removed from the online change state.
XRFALT	LCL	XRF alternate system.

Return, Reason, and Completion Codes for QUERY MEMBER

The return and reason codes that can be returned as a result of the QUERY MEMBER command are standard for all commands entered through the OM API. Refer to the OM Return and Reason code section in the *IMS Version 9: Common Service Layer Guide and Reference* for the list of codes and their meanings.

Table 117 contains the return and reason codes that can be returned to OM for a QUERY MEMBER command. It also includes a brief explanation of the codes.

Table 117. Return and Reason Codes for QUERY MEMBER Command

Return Code	Reason Code	Meaning
X'00000000'	X'00000000'	The QUERY MEMBER TYPE(IMS) command completed successfully.
X'00000014'	X'00005004'	The QUERY MEMBER command failed because a DFSOCMD response buffer could not be obtained.

Errors unique to the processing of this command are returned as a completion codes. A completion code is returned for each action against an individual member.

Table 118 contains the completion code that can be returned on a QUERY MEMBER command.

Table 118. Completion Codes for QUERY MEMBER

Completion Code	Meaning
0	The QUERY MEMBER TYPE(IMS) command completed successfully for this IMS.

Examples for QUERY MEMBER Command

This section provides OM API and TSO SPOC input and output examples and explanations for the QUERY command.

Example 1 for QUERY MEMBER TYPE(IMS) Command

TSO SPOC input:

```
QRY MEMBER TYPE(IMS) SHOW(ALL)
```

TSO SPOC output:

```
MbrName    CC Type  Status LclAttr LclStat    ModId
SYS3        0 IMS                                1
```

OM API input:

```
CMD(QRY MEMBER TYPE(IMS) SHOW(ALL))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2002.163 16:32:12.998765</statime>
<stotime>2002.163 16:32:12.999775</stotime>
<staseq>B7C4B78AFD86D562</staseq>
<stoseq>B7C4B78AFDC5FA80</stoseq>
<rqsttkn1>USRT002 10093212</rqsttkn1>
<rc>00000000</rc>
```

```
<rsn>00000000</rsn>
</ctl>
<cmd>
<master>SYS3    </master>
<userid>USRT002 </userid>
<verb>QRY </verb>
<kwd>MEMBER  </kwd>
<input>QRY MEMBER TYPE(IMS) SHOW(ALL)</input>
</cmd>
<cmdrsphdr>
<hdr slbl="MBR"   llbl="MbrName" scope="LCL" sort="a" key="1" scroll="no"  len="8"
dtype="CHAR" align="left" />
<hdr slbl="CC"    llbl="CC"      scope="LCL" sort="n" key="0" scroll="yes" len="4"
dtype="INT" align="right" />
<hdr slbl="TYP"   llbl="Type"    scope="LCL" sort="n" key="0" scroll="yes" len="5"
dtype="CHAR" align="left" />
<hdr slbl="STT"   llbl="Status"  scope="GBL" sort="n" key="0" scroll="yes" len="*"
dtype="CHAR" align="left" />
<hdr slbl="LATTR" llbl="LclAttr" scope="LCL" sort="n" key="0" scroll="yes" len="*"
dtype="CHAR" align="left" />
<hdr slbl="LSTT"  llbl="LclStat" scope="LCL" sort="n" key="0" scroll="yes" len="*"
dtype="CHAR" align="left" />
<hdr slbl="MODI"  llbl="ModId"   scope="LCL" sort="n" key="0" scroll="yes" len="8"
dtype="CHAR" align="right" /></cmdrsphdr>
<cmdrspdata>
<rsp>MBR(SYS3    ) CC(   0) TYP(IMS) MODI(       1) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: IMS member SYS3 is active in the IMSplex.

Example 2 for QUERY MEMBER TYPE(IMS) Command

TSO SPOC input:

```
QRY MEMBER TYPE(IMS) SHOW(ALL)
```

TSO SPOC output:

```
MbrName    CC Type  Status LclAttr LclStat    ModId
SYS3        0 IMS                 LEOPT          1
```

OM API input:

```
CMD(QRY MEMBER TYPE(IMS) SHOW(ALL))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2002.163 16:42:10.557119</statime>
<stotime>2002.163 16:42:10.557503</stotime>
<staseq>B7C4B9C4DDCBF28D</staseq>
<stoseq>B7C4B9C4DDE3F02D</stoseq>
<rqsttkn1>USRT002 10094210</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<master>SYS3    </master>
<userid>USRT002 </userid>
<verb>QRY </verb>
<kwd>MEMBER </kwd>
<input>QRY MEMBER TYPE(IMS) SHOW(ALL) </input>
</cmd>
<cmdrsphdr>
```

```
<hdr slbl="MBR"   llbl="MbrName" scope="LCL" sort="a" key="1" scroll="no"  len="8"
dtype="CHAR" align="left" />
<hdr slbl="CC"    llbl="CC"      scope="LCL" sort="n" key="0" scroll="yes" len="4"
dtype="INT" align="right" />
<hdr slbl="TYP"   llbl="Type"    scope="LCL" sort="n" key="0" scroll="yes" len="5"
dtype="CHAR" align="left" />
<hdr slbl="STT"   llbl="Status"  scope="GBL" sort="n" key="0" scroll="yes" len="*"
dtype="CHAR" align="left" />
<hdr slbl="LATTR" llbl="LclAttr" scope="LCL" sort="n" key="0" scroll="yes" len="*"
dtype="CHAR" align="left" />
<hdr slbl="LSTT"  llbl="LclStat" scope="LCL" sort="n" key="0" scroll="yes" len="*"
dtype="CHAR" align="left" />
<hdr slbl="MODI"  llbl="ModId"   scope="LCL" sort="n" key="0" scroll="yes" len="8"
dtype="CHAR" align="right" />
</cmdrsphdr>
<cmdrspdata>
<rsp>MBR(SYS3    ) CC(   0) TYP(IMS) LSTT(LEOPT) MODI(       1) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: IMS member SYS3 is active in the IMSplex. The local status shows that LE runtime option overrides are enabled.

Example 3 for QUERY MEMBER TYPE(IMS) Command

TSO SPOC input:

```
QRY MEMBER TYPE(IMS) SHOW(ALL)
```

TSO SPOC output:

```
 Response for: QUERY MEMBER TYPE(IMS) SHOW(ALL)
MbrName    CC Type  Status LclAttr        LclStat    ModId
IMS2        0 IMS          SHAREDQ,GBLOLC                1
IMS3        0 IMS          SHAREDQ,GBLOLC                1
SYS3        0 IMS          SHAREDQ,GBLOLC                1
```

OM API input:

```
CMD (QRY MEMBER TYPE(IMS) SHOW(ALL))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2002.163 15:13:05.255654</statime>
<stotime>2002.163 15:13:06.479196</stotime>
<staseq>B7C4A5DB308E6544</staseq>
<stoseq>B7C4A5DC5B45C385</stoseq>
<rqsttkn1>USRT011 10081304</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<master>IMS3 </master>
<userid>USRT011</userid>
<verb>QRY</verb>
<kwd>MEMBER</kwd>
<input>QUERY MEMBER TYPE(IMS) SHOW(ALL)</input>
</cmd>
<cmdrsphdr>
<hdr slbl="MBR"   llbl="MbrName" scope="LCL" sort="a" key="1" scroll="no"  len="8"
dtype="CHAR" align="left" />
<hdr slbl="CC"    llbl="CC"      scope="LCL" sort="n" key="0" scroll="yes" len="4"
dtype="INT"  align="right" />
```

```
<hdr slbl="TYP"   llbl="Type"    scope="LCL" sort="n" key="0" scroll="yes" len="5"
dtype="CHAR" align="left" />
<hdr slbl="STT"   llbl="Status"  scope="GBL" sort="n" key="0" scroll="yes" len="*"
dtype="CHAR" align="left" />
<hdr slbl="LATTR" llbl="LclAttr" scope="LCL" sort="n" key="0" scroll="yes" len="*"
dtype="CHAR" align="left" />
<hdr slbl="LSTT"  llbl="LclStat" scope="LCL" sort="n" key="0" scroll="yes" len="*"
dtype="CHAR" align="left"/>
<hdr slbl="MODI"  llbl="ModId"   scope="LCL" sort="n" key="0" scroll="yes" len="8"
dtype="CHAR" align="right"/>
</cmdrsphdr>
<cmdrspdata>
<rsp>MBR(IMS3    ) CC(   0) TYP(IMS) LATTR(SHAREDQ,GBLOLC) MODI(       1) </rsp>
<rsp>MBR(IMS2    ) CC(   0) TYP(IMS) LATTR(SHAREDQ,GBLOLC) MODI(       1) </rsp>
<rsp>MBR(SYS3    ) CC(   0) TYP(IMS) LATTR(SHAREDQ,GBLOLC) MODI(       1) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: IMS members SYS3, IMS2, and IMS3 are active in the IMSplex. All three IMSs have shared queues enabled and global online change enabled.

QUERY OLC

For an IMS running with RM (RMENV=Y), the QUERY OLC command returns information about the OLCSTAT DS that is shared by all of the IMS systems participating in global online change. The command response is the same for all of the IMS systems. QUERY OLC is valid for an IMS enabled for global online change but it is not valid for an IMS enabled for local online change. QUERY OLC is not supported on an XRF alternate, an RSR tracker, or an FDR region. It can only be specified through the OM API.

The QUERY OLC LIBRARY command displays information about global online change, such as the current online change libraries and the IMSs that are current with the online change libraries. QUERY OLC LIBRARY(OLCSTAT) displays the contents of the global online change status data set, OLCSTAT. Specifying the SHOW keyword may optionally show the current active online change libraries, the list of IMSs that are current with the online change libraries, the modify id, and the last online change that was done.

For an IMS running without RM services (RMENV=N), the QUERY OLC command returns information about the local OLCSTAT DS of an IMS system. The command response is different for each IMS because each IMS is required to have a unique OLCSTAT DC. In a no RM environment, if more than one IMS is specified in the route list for the QUERY OLC, only the OLCSTAT DS information for the command IMS master is returned. To obtain OLCSTAT DS information from each IMS that is running without RM, the QUERY OLC command must be issued separately to each IMS. To determine which IMSs are defined with RMENVNO, issue a QUERY MEMBER SHOW(ATTRIB) command.

The command syntax for this command is defined in XML and is available to automation programs which communicate with OM.

LIBRARY
Online change library. The library can be the following:

OLCSTAT
OLCSTAT data set contents.

SHOW()
Specifies the library information returned.

ALL Returns all the output fields.

ACTVLIB
Displays the suffixed online change library names that are currently active. This includes ACBLIBA or ACBLIBB, FMTLIBA or FMTLIBB, and MODBLKSA or MODBLKSB. These are the online change libraries the IMS online system must use at IMS initialization time.

DSN OLCSTAT data set name.

LASTOLC
Displays the last online change that was successfully performed. If an IMS was down during the last online change and its restart type does not conflict with the last online change that was performed, it will be permitted to warmstart. The last online change type is blank, if no online changes have been done. The last online change type is one or more of the following, if at least one online change has been done:

- ACBLIB
- FMTLIB
- MODBLKS

MBRLIST
Displays the list of IMSs that are current with the online change libraries. These are the IMSs that will be permitted to warmstart. The IMSs either participated in the last online change, or coldstarted since the last online change.

MODID
Modify id. The modify id -1 represents the number of global online changes that have been performed.

QUERY OLC Output Fields

Table 119 shows the QUERY OLC output fields. The columns in the table are as follows:

Short Label
Contains the short label generated in the XML output.

Keyword
Identifies the keyword on the command that caused the field to be generated. N/A appears for output fields that are always returned.

Scope Identifies the scope of the output field.

Meaning
Provides a brief description of the output field.

Table 119. Output Fields for the QUERY OLC Command

Short Label	Keyword	Scope	Meaning
MBR	N/A	N/A	IMSplex member that built output line. IMS identifier of the IMS that build the output. IMS identifier is always returned.
CC	N/A	N/A	Completion code for the line of output. The completion code indicates whether or not IMS was able to process the command for the specified library. Refer to Table 121 on page 513 for more information. The completion code is always returned.

Table 119. Output Fields for the QUERY OLC Command (continued)

Short Label	Keyword	Scope	Meaning
LIB	N/A	GBL	Library name. Can be OLCSTAT.
ACBL	ACTVLIB	GBL	Current ACBLIB library. A means the current ACBLIB library is ACBLIBA. B means the current ACBLIB library is ACBLIBB.
FMTL	ACTVLIB	GBL	Current FMTLIB library. A means the current FMTLIB library is FMTLIBA. B means the current FMTLIB library is FMTLIBB.
MODB	ACTVLIB	GBL	Current MODBLKS and MATRIX libraries. A means the current libraries are MODBLKSA and MATRIXA. B means the current libraries are MODBLKSB and MATRIXB.
DSN	DSN	GBL	OLCSTAT data set name.
LAST	LASTOLC	GBL	The last successful online change that was successfully performed. If an IMS was down during the last online change and its restart type does not conflict with the last online change that was performed, it will be permitted to warmstart. The online change type may include one or more of the following: • ACBLIB • FMTLIB • MODBLKS
MBRL	MBRLIST	GBL	List of IMSplex members that are current with the online change libraries. These are the IMSs that will be permitted to warmstart. The IMSs either participated in the last online change, or coldstarted since the last online change.
MODI	MODID	GBL	Current modify id. The modify id - 1 is the number of successful global online changes that have been performed.

Return, Reason, and Completion Codes for QUERY OLC

The return and reason codes that can be returned as a result of the QUERY OLC command are standard for all commands entered through the OM API. Refer to the OM Return and Reason code section in the *IMS Version 9: Common Service Layer Guide and Reference* for the list of codes and their meanings.

Table 120 on page 513 contains the return and reason codes that can be returned to OM from a QUERY OLC command.

Table 120. Return and Reason Codes for the QUERY OLC Command

Return Code	Reason Code	Meaning
X'00000000'	X'00000000'	The QUERY OLC command completed successfully.
X'00000004'	X'00001000'	The QUERY OLC command was not processed on the IMS system as the IMS system is not the command master. No information is returned.
X'00000010'	X'0000400C'	The QUERY OLC command failed because it is invalid for an XRF alternate.
X'00000010'	X'00004014'	The QUERY OLC command failed because it is invalid for an RSR tracker.
X'00000010'	X'0000401C'	The QUERY OLC command failed because it is invalid for an FDR region.
X'00000010'	X'0000410C'	The QUERY OLC command is rejected, because global online change is not enabled. Local online change is enabled. Use the /DISPLAY MODIFY command for local online change. If your IMSplex is made up of some IMSs that support global online change and some that support local online change, route the QUERY OLC command to an IMS that is enabled for global online change. Issue the QUERY MEMBER TYPE(IMS) SHOW(ATTRIB) command to choose an IMS that has global online change enabled.
X'00000010'	X'00004114'	The QUERY OLC LIBRARY(OLCSTAT) command failed because of an error accessing the OLCSTAT data set. A DFS2843 message is sent to the OM output exit as unsolicited output.
X'00000010'	X'00004118'	The QUERY OLC LIBRARY(OLCSTAT) command failed because of an error allocating the OLCSTAT data set. A DFS2848 message is sent to the OM output exit as unsolicited output.
X'00000010'	X'0000411C'	The QUERY OLC LIBRARY(OLCSTAT) command failed because of an error in the OLCSTAT data set contents. One or more of the values is invalid. A DFS2844 message is sent to the OM output exit as unsolicited output.
X'00000014'	X'00005004'	The QUERY OLC command processing failed because a DFSOCMD response buffer could not be obtained.
X'00000014'	X'00005FFF'	The QUERY OLC command failed because of an internal IMS error.

Errors unique to the processing of this command are returned as a completion codes. A completion code is returned for each action against an individual library.

Table 121 contains the completion codes that can be returned on a QUERY OLC command.

Table 121. Completion Codes for the QUERY OLC Command

Completion Code	Meaning
0	The QUERY OLC command completed successfully for the library.

Examples for QUERY OLC Command

This section provides OM API and TSO SPOC input and output examples and explanations for the QUERY command.

Example 1 for QUERY OLC Command

TSO SPOC input:

```
QRY OLC LIBRARY(OLCSTAT) SHOW(ACTVLIB,MODID,MBRLIST)
```

TSO SPOC output:

```
 Response for: QUERY OLC LIBRARY(OLCSTAT) SHOW(ACTVLIB,MODID,MBRLIST)
MbrName    CC Library    ACBLIB   FMTLIB  MODBLKS    Modid MbrList
MS3         0 OLCSTAT         B        A        B        1 IMS3,IMS2,SYS3
```

OM API input:

```
CMD (QRY OLC LIBRARY(OLCSTAT) SHOW(ACTVLIB,MODID,MBRLIST))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2002.163 15:22:43.468642</statime>
<stotime>2002.16315:22:45.400709</stotime>
<staseq>B7C4A8029DD62884</staseq>
<stoseq>B7C4A80475885248</stoseq>
<rqsttkn1>USRT011 10082243</rqsttkn1>
<rc>0200000C</rc>
<rsn>00003000</rsn>
</ctl>
<cmderr>
<mbr name="IMS2    ">
<typ>IMS     </typ>
<styp>DBDC</styp>
<rc>00000004</rc>
<rsn>00001000</rsn>
</mbr>
<mbr name="SYS3    ">
<typ>IMS     </typ>
<styp>DBDC   </styp>
<rc>00000004</rc>
<rsn>00001000</rsn>
</mbr>
</cmderr>
<cmd>
<master>IMS3    </master>
<userid>USRT011 </userid>
<verb>QRY </verb>
<kwd>OLC  </kwd>
<input>QUERY OLC LIBRARY(OLCSTAT) SHOW(ACTVLIB,MODID,MBRLIST)</input>
</cmd>
<cmdrsphdr>
<hdr slbl="MBR"  llbl="MbrName" scope="LCL" sort="a" key="1" scroll="no" len="8"
dtype="CHAR" align="left" />
<hdr slbl="CC"   llbl="CC"      scope="LCL" sort="n" key="0" scroll="yes" len="4"
dtype="INT"  align="right" />
<hdr slbl="LIB"  llbl="Library" scope="GBL" sort="a" key="1" scroll="no"  len="8"
dtype="CHAR" align="left" />
<hdr slbl="ACBL" llbl="ACBLIB"  scope="GBL" sort="n" key="0" scroll="yes" len="8"
dtype="CHAR" align="right" />
<hdr slbl="FMTL" llbl="FMTLIB"  scope="GBL" sort="n" key="0" scroll="yes" len="8"
dtype="CHAR" align="right" />
<hdr slbl="MODB" llbl="MODBLKS" scope="GBL" sort="n" key="0" scroll="yes" len="8"
```

```
dtype="CHAR" align="right" />
<hdr slbl="MODI" llbl="Modid"   scope="GBL" sort="n" key="0" scroll="yes" len="8"
dtype="CHAR" align="right" />
<hdr slbl="MBRL" llbl="MbrList" scope="GBL" sort="n" key="0" scroll="yes" len="*"
dtype="CHAR" align="left"/>
</cmdrsphdr>
<cmdrspdata>
<rsp>MBR(IMS3    ) CC(   0) LIB(OLCSTAT ) ACBL(B) FMTL(A) MODB(B) MODI(       1)
MBRL(IMS3,IMS2,SYS3) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: `QUERY OLC LIBRARY(OLCSTAT)` displays the contents of the OLCSTAT data set, which contains global online change status. This command displays the active online change libraries, the modify id, and the list of IMS members that are current with the online change libraries and may therefore warm list. The output shows that the ACBLIBB data set is active, the FMTLIBB data set is active, the MODBLKSB data set is active, and the modify id is 2. SYS3 was the command master that built the output.

Example 2 for QUERY OLC Command

TSO SPOC input:

```
QRY OLC LIBRARY(OLCSTAT) SHOW(DSN)
```

TSO SPOC output:

```
MbrName    CC Library  DSName
IMS3        0 OLCSTAT  IMSTESTL.IMS02.OLCSTAT
```

OM API input:

```
CMD (OLC LIBRARY(OLCSTAT) SHOW(DSN))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2002.163 15:28:36.353742</statime>
<stotime>2002.16315:28:36.426823</stotime>
<staseq>B7C4A953276CE286</staseq>
<stoseq>B7C4A95339447348</stoseq>
<rqsttkn1>USRT011 10082836</rqsttkn1>
<rc>0200000C</rc><rsn>00003000</rsn>
</ctl>
<cmderr>
<mbr name="IMS2    ">
<typ>IMS     </typ>
<styp>DBDC    </styp>
<rc>00000004</rc>
<rsn>00001000</rsn>
</mbr>
<mbr name="SYS3    ">
<typ>IMS     </typ>
<styp>DBDC    </styp>
<rc>00000004</rc>
<rsn>00001000</rsn>
</mbr>
</cmderr>
<cmd>
<master>IMS3    </master>
<userid>USRT011 </userid>
<verb>QRY </verb>
```

```
<kwd>OLC  </kwd>
<input>QUERY OLC LIBRARY(OLCSTAT) SHOW(DSN)</input>
</cmd>
<cmdrsphdr>
<hdr slbl="MBR" llbl="MbrName" scope="LCL" sort="a" key="1" scroll="no"  len="8"
dtype="CHAR" align="left" />
<hdr slbl="CC"  llbl="CC"      scope="LCL" sort="n" key="0" scroll="yes" len="4"
dtype="INT"  align="right" />
<hdr slbl="LIB" llbl="Library" scope="GBL" sort="a" key="1" scroll="no"  len="8"
dtype="CHAR" align="left" />
<hdr slbl="DSN" llbl="DSName"  scope="GBL" sort="n" key="0" scroll="yes" len="8"
dtype="CHAR" align="left" />
</cmdrsphdr>
<cmdrspdata>
<rsp>MBR(IMS3   ) CC(   0) LIB(OLCSTAT ) DSN(IMSTESTL.IMS02.OLCSTAT) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: This QUERY OLC command displays the OLCSTAT data set name. IMS3 was the command master that built the output.

QUERY STRUCTURE

The QUERY STRUCTURE command displays information about IMS coupling facility structure(s) used by members of an IMSplex.

In IMS Version 9, this command is supported only by RM to return information about a resource structure.

The command syntax for this command is defined in XML and is available to automation programs which communicate with OM.

NAME()
Specifies the names of the structures for which information is to be returned. The structure name may be a generic parameter, to allow easy specification of a group of structures whose names match a generic parameter mask.

SHOW()
Specifies the output fields to be returned. If SHOW is not specified, only the structure names, IMSplex member that builds the output line, and completion codes are returned. This provides a method for a system management application to obtain a list of all structure names. This can be used to determine the resource structure that is managed by RM. The parameters supported with the SHOW keyword are as follows:

ALL Returns all the output fields.

STATISTICS
Displays statistics information for the structures that match the specification in the NAME() parameter.

TYPE Returns the type of the specified structure. For example, the type may be RSRC, which identifies a resource structure.

QUERY STRUCTURE Output Fields

Table 122 on page 517 contains information about the output fields for QUERY STRUCTURE. The columns in the table are as follows:

Short Label
Contains the short label generated in the XML output.

Keyword

Identifies the keyword on the command that caused the field to be generated. N/A appears for output fields that are always returned.

Meaning

Provides a brief description of the output field.

Table 122. Output Fields for the QUERY STRUCTURE Command

Short Label	Keyword	Meaning
STRNM	N/A	Resource structure name. The structure name is always returned.
MBR	N/A	IMSplex member that built the output line. The RM identifier of the RM that built the output line.
CC	N/A	Completion code for the line of output. The completion code is always returned.
TYP	TYPE	Structure type. In IMS Version 9, only resource type RSRC is supported.
LEA	STATISTICS	Number of list entries that are allocated in the structure.
LEI	STATISTICS	Number of list entries in use in the structure.
ELMA	STATISTICS	Number of data elements that are allocated in the structure.
ELMI	STATISTICS	Number of data elements in use in the structure.
RATIO	STATISTICS	Entry to element ratio. It is in the format of list entries/data elements.

Return, Reason, and Completion Codes for QUERY STRUCTURE

The return and reason codes that can be returned as a result of the QUERY STRUCTURE command are standard for all commands entered through the OM API. Refer to the OM Return and Reason code section in the *IMS Version 9: Common Service Layer Guide and Reference* for the list of codes and their meanings.

Table 123 on page 519 contains the return and reason codes that can be returned to OM from a QUERY STRUCTURE command.

Table 123. Return and Reason Codes for the QUERY STRUCTURE Command

Return Code	Reason Code	Meaning
X'00000000'	X'00000000'	The QUERY STRUCTURE command completed successfully.
X'0300000C'	X'00003000'	The QUERY STRUCTURE command is successful for at least one resource name. The QUERY STRUCTURE command is not successful for one or more resource names. The Completion Code indicates the reason for the error with the resource name. The Completion Codes that can be returned by the QUERY STRUCTURE command are listed in the QUERY STRUCTURE completion code table.
X'0300000C'	X'00003004'	No requests were successful. The resource name(s) specified may be invalid or there were no resources that match the filter specified.
X'03000014'	X'0000502C'	The QUERY STRUCTURE command processing terminated. RM was unable to obtain storage for the command output header.
X'03000014'	X'00005030'	The QUERY STRUCTURE command processing terminated. RM was unable to obtain storage for the command output response.
X'03000014'	X'00005200'	The QUERY STRUCTURE command processing terminated due to an unexpected CQS error.

Errors unique to the processing of this command are returned as a completion codes. A completion code is returned for each action against an individual resource.

Table 124 contains the completion codes that can be returned on a QUERY STRUCTURE command.

Table 124. Completion Codes for the QUERY STRUCTURE Command

Completion Code	Meaning
0	The QUERY STRUCTURE command completed successfully for the resource.
4	The structure name is unknown to the client that is processing the request. The structure name may have been typed in error or the structure may not be defined or allocated at this time. If this is a wildcard request there were no matches for the name. Confirm the correct spelling of the structure name is specified on the command.

Example for QUERY STRUCTURE

This section provides OM API and TSO SPOC input and output examples and explanations for the QUERY command.

Example 1 for QUERY STRUCTURE Command

TSO SPOC input:

```
QRY STRUCTURE SHOW(STATISTICS)
```

TSO SPOC output:

```
StructureName    MbrName    CC LeAlloc LeInuse ElmAlloc ElmInuse LE/EL
IMSRSRC01        RM1RM       0    3577     676     3574       24 0001/0001
```

OM API input:

```
CMD (QRY STRUCTURE SHOW(STATISTICS))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1</xmlvsn>
<statime>2002.16314:31:34.901057</statime>
<stotime>2002.16314:31:34.941134</stotime>
<staseq>B7C49C9433D410C1</staseq>
<stoseq>B7C49C943D9CEC44</stoseq>
<rqsttkn1>USRT011 10073134</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<master>RM1RM   </master>
<userid>USRT011</userid>
<verb>QRY </verb>
<kwd>STRUCTURE</kwd>
<input>QUERY STRUCTURE SHOW(ALL)</input>
</cmd>
<cmdrsphdr>
<hdr slbl="STRNM" llbl="StructureName" scope="LCL" sort="A" key="1" scroll="NO"
len="16" dtype="CHAR" align="left"/>
<hdr slbl="MBR"   llbl="MbrName"  scope="LCL" sort="N" key="0" scroll="NO" len="8"
dtype="CHAR" align="left" />
<hdr slbl="CC"    llbl="CC"       scope="LCL" sort="N" key="0" scroll="YES" len="4"
dtype="INT"  align="right" />
<hdr slbl="TYP"   llbl="Type"     scope="LCL" sort="N" key="0" scroll="YES" len="8"
dtype="CHAR" align="left" />
<hdr slbl="LEA"   llbl="LeAlloc"  scope="LCL" sort="N" key="0" scroll="YES" len="4"
dtype="INT"  align="right" />
<hdr slbl="LEI"   llbl="LeInuse"  scope="LCL" sort="N" key="0" scroll="YES" len="4"
dtype="INT"  align="right" />
<hdr slbl="ELMA"  llbl="ElmAlloc" scope="LCL" sort="N" key="0" scroll="YES" len="4"
dtype="INT"  align="right" />
<hdr slbl="ELMI"  llbl="ElmInuse" scope="LCL" sort="N" key="0" scroll="YES" len="4"
dtype="INT"  align="right" />
<hdr slbl="RATIO" llbl="LE/EL"    scope="LCL" sort="N" key="0" scroll="YES" len="9"
dtype="CHAR" align="left" />
</cmdrsphdr>
<cmdrspdata>
<rsp>STRNM(IMSRSRC01) MBR(RM1RM) CC(   0) TYP(RSRC) LEA(3577) LEI( 676) ELMA(3574)
ELMI(  24) RATIO(0001/0001)</rsp>
</cmdrspdata>
</imsout>
```

Explanation: This command displays all of the resource structures in the IMSplex and their statistics. Only one resource structure is defined, IMSRSRC01. The list entries allocated on the resource structure is 3577, the list entries in use on the structure is 680. The data elements allocated is 3574, the data elements in use is 32. The list entry to data element ratio on the resource structure is one to one. Not many list entries or data elements are in use on the resource structure, so the resource structure is not approaching full.

QUERY TRAN

The QUERY TRAN command displays information about transactions (for example, class, status, queue count and others) across the IMSplex. This command can be specified only through the OM API and is valid on an XRF alternate.

The transaction information displayed depends if the IMS issuing the QUERY TRAN command is running with RM services or not. If QUERY TRAN is issued by an IMS command master running without RM, all transaction information local to that IMS is returned. If QUERY TRAN is issued by an IMS command master running with RM, the IMS command master retreives global information from CQS or RM as specified.

If QUERY TRAN is routed for global information to all IMSs in an environment where some IMSs use RM services and other IMSs do not, the command results will vary because any of the IMS systems can be the command master. The RM environment of the IMS command master affects the type of transaction information that is displayed. You may want to route QUERY TRAN to specific IMS systems if some IMSs are using RM. Below are two examples of why you may receive different command results:

1. The IMSplex has non-cloned systems and the transaction is only defined by an IMS that has RMENV=N. The IMS command master is running with RM services, but because the transaction is not defined to RM, no global information is obtained. The results are two response lines:
   - The IMS command master returns the transaction name as invalid.
   - The IMS with RMENV=N returns its local information.
2. The IMSplex has non-cloned systems. If global queue counts are requested, and the IMS command master does not have RM running, no global queue counts are returned for transactions that are not defined locally at the command master. All other IMSs return only their local information.

CLASS()

Displays transactions that possess at least one of the specified classes. This allows for additional filtering by CLASS value. If a STATUS, CLASS, or QCNT filter is not specified, all of the transactions matching the transaction name are returned.

The output returned when the CLASS filter is specified includes the class value of the transaction that caused the transaction name to be displayed even if the SHOW(CLASS) option is not specified.

NAME()

Displays the transactions or group of transactions whose name matches the generic or wildcard parameter specified. If a STATUS, CLASS, or QCNT filter is not specified, all of the transactions matching the transaction name are returned.

QCNT()

Selects transactions that have a queue count less than (LT), less than or equal to (LE), greater than (GT), greater than or equal to (GE), equal to (EQ), or not equal to (NE) the specified numbers. The specified number cannot be a 1 when LT is specified. This allows additional filtering by QCNT value.

Transactions with a queue count of 0 are not returned when the QCNT filter is specified. When a filter of QCNT(LT,n) is specified, transactions with a queue count greater than 0 and less than 'n' are returned. If a STATUS, CLASS, or QCNT filter is not specified, all of the transactions matching the transaction name are returned.

The QCNT filter is valid in both a shared queues environment and a non shared queues environment.

In a shared queues environment, if QCNT is specified, the performance implication is that the shared queues are read. In this environment, the QRY

TRAN QCNT command is processed only by the master IMS as the queues are global. The command master returns all the transactions on the shared queues that match the queue count filter specified. If QCNT is specified with a wildcard transaction name, the performance implication is that all of the shared queues transaction messages on the Coupling Facility must be read.

In a non shared queues environment, the local queue count values are used to determine the transactions to be displayed. In this environment, the QRY TRAN QCNT command is processed by each IMS the command is routed to as the queues are local. Each IMS returns all the transactions it found locally that match the queue count filter specified.

The output returned when the QCNT filter is specified includes the queue count of the transaction that caused the transaction name to be displayed even if the SHOW(QCNT) option is not specified.

SHOW()

Specifies the transaction output fields to be returned. The transaction name is always returned along with the name of the IMS that created the output for the transaction and the completion code. If SHOW is not specified, only the transaction names are returned provided the QCNT, CLASS or STATUS filter is not specified. This provides a method for a system management application to obtain a list of transactions matching the transaction name that are currently known in the IMSplex.

ALL Returns all the output fields both local and global values except the affinity of the messages on the shared queues. This affinity that is shown when the AFFIN option is specified is valid only with the QCNT filter and is ignored for the other flavors of the QRY TRAN command. Global values are only returned for those status fields and attributes for which global information is kept for the IMSplex. In IMS Version 9, only the QCNT attribute may have both local and global values.

AFFIN IMS affinity of the messages on the shared queues. The only SHOW option supported when the QCNT() filter is specified is the AFFIN option. No other SHOW options are supported with the QCNT() filter due to performance reasons.

AFFIN is valid only in a shared queues environment and is ignored in a non-shared queues environment.

CLASS
Scheduling class used to determine which message regions can process the transaction locally on a particular IMS.

CPRI Local current scheduling priority.

LCT Limit count in the local IMS.

LPRI Local limit scheduling priority.

MAXRGN
Local maximum region count.

NPRI Local normal scheduling priority.

PARLIM
Local parallel processing limit count. If load balancing is disabled a numeric value of 65535 is returned. If load balancing is enabled, the PARLIM value between 0-37327 is returned.

PLCT Local processing limit count.

PSB PSB name associated with the transaction.

QCNT Local transaction message queue count.

RGC Number of regions the transaction is currently scheduled in the local IMS.

SEGNO
Local application program output segment limit allowed in the message queues for each GU call.

SEGSZ
Local application program output segment size limit allowed in the message queues for each GU call.

STATUS
Local transaction status.

STATUS()
Selects transactions for display that possess at least one of the specified transaction status. This allows for additional filtering by transaction status. If a STATUS, CLASS, or QCNT filter is not specified, all of the transactions matching the transaction name are returned.

The output returned when the STATUS filter is specified includes the status of the transaction that caused the transaction name to be displayed even if the SHOW(STATUS) option is not specified.

RM does not keep any global status for transactions, so all status is local. In a follow-on release, RM will keep certain types of global transaction status. At that time, filtering by transaction status will filter on both local and global status.

BAL Transaction is eligible for load balancing (for example, with parallel limits specified).

CONV This is a conversational transaction.

CPIC This CPI-C transaction was built dynamically on this IMS system and can process only on this IMS system.

DYN Transaction was built in a shared queues environment, is not defined to this IMS, and therefore, cannot be scheduled in this IMS subsystem.

FPE This transaction is Fast Path exclusive.

FPP This transaction is Fast Path potential.

IOPREV
Indicates that a BMP program containing GSAM cannot completed scheduling because I/O prevention has not completed. Further I/O requests to data sets are inhibited.

LCK Transaction locked by a `/LOCK TRANSACTION` command.

QERR I/O error has occurred on this queue for this MSC remote transaction.

RESP Response mode transaction.

RMT Remote transaction.

SUSPEND
Transaction has messages on the suspend queue.

STOQ Transaction is stopped for queueing. This might be due to a previous `UPDATE TRAN`, `/PURGE TRAN` or `/STO TRAN` command.

STOSCHD
Transaction is stopped for scheduling. This might be due to a previous `UPDATE TRAN`, `/PSTOP TRAN` or `/STO TRAN` command or an application abend.

TRACE
Transaction is being traced.

USTO Transaction scheduling stopped because of unavailable data.

QUERY TRAN Compared to Other Commands

Table 125 shows different instances of the `QUERY TRAN` command and other IMS commands that display similar information.

Table 125. QUERY TRAN Compared to Other Similar Commands

QUERY TRAN Command	Similar IMS Command
`QUERY TRAN NAME(tranname) SHOW(ALL)`	`/DISPLAY TRAN tranname`
`QUERY TRAN SHOW(ALL)`	`/DISPLAY TRAN ALL`
`QUERY TRAN NAME(tranname) SHOW(QCNT)`	`/DISPLAY TRAN tranname QCNT`
`QUERY TRAN NAME(tranname) STATUS(IOPREV,LCK,QERR,SUSPEND,STOQ, STOSCHD,USTO) SHOW(STATUS)`	`/DISPLAY STATUS TRANSACTION`

QUERY TRAN Output Fields

Table 126 shows the output fields for the `QUERY TRAN` command. The columns in the table are as follows:

Short Label
Contains the short label generated in the XML output.

Keyword
Identifies the keyword on the command that caused the field to be generated. N/A appears for output fields that are always returned.

Scope Identifies the scope of the output field.

Meaning
Provides a brief description of the output field.

Table 126. Output Fields for the QUERY TRAN Command

Short Label	Keyword	Scope	Meaning
TRAN	N/A	N/A	Transaction name. A transaction defines the processing characteristics of messages destined for an application program. The transaction name is always returned.
MBR	N/A	N/A	IMSplex member that built the output line. IMS identifier of IMS that built the output. The IMS identifier is always returned.

Table 126. Output Fields for the QUERY TRAN Command (continued)

Short Label	Keyword	Scope	Meaning
CC	N/A	N/A	Completion code. The completion code indicates whether or not IMS was able to process the command for the specified resource. The completion code is always returned. Refer Table 128 on page 529 for more information.
AFIN	AFFIN	GBL	Affinity of the messages on the shared queues. AFFIN displays the IMSid/RSEname of the IMS system that the message can be processed on.
LCLS	CLASS	LCL	Scheduling class used to determine which message regions can process the transaction locally on a particular IMS.
LCP	CPRI	LCL	Local current scheduling priority. The current scheduling priority is used to calculate which transaction is selected for scheduling.
LLCT	LCT	LCL	Limit count in the local IMS. The limit count is the number that, when compared to the number of input transactions queued and waiting to be processed, determines whether the normal or limit priority value is assigned to this transaction.
LLP	LPRI	LCL	Local limit scheduling priority. The limit scheduling priority is the priority to which this transaction is raised when the number of input transactions enqueued and waiting to be processed is equal to or greater than the limit count value.
LMRG	MAXRGN	LCL	Local maximum region count. The maximum region count is the maximum number of message processing program (MPP) regions that can be concurrently scheduled to process a transaction that is eligible for parallel scheduling.
LNP	NPRI	LCL	Local normal scheduling priority. The normal scheduling priority is the priority assigned to this transaction when the number of input transactions enqueued and waiting to be processed is less than the limit count value.
LPLM	PARLIM	LCL	Local parallel processing limit count. The parallel limit count is the maximum number of messages that can currently be queued, but not yet processed, by each active message region currently scheduled for this transaction. An additional message region is scheduled whenever the transaction queue count exceeds the PARLIM value multiplied by the number of regions currently scheduled for this transaction.
LPLCT	PLCT	LCL	Local processing limit count. The processing limit count is the number of transaction messages a program can process in a single scheduling.
PSB	PSB	LCL	PSB name associated with the transaction.
Q	QCNT	GBL	Global transaction message queue count on the shared queues. Q is only displayed if shared queues are used.

Table 126. Output Fields for the QUERY TRAN Command (continued)

Short Label	Keyword	Scope	Meaning
LQ	QCNT	LCL	Local transaction message queue count.
RGC	RGC	LCL	Number of regions the transaction is currently scheduled in the local IMS.
LSNO	SEGNO	LCL	Local application program output segment limit allowed in message queues for each GU call.
LSSZ	SEGSZ	LCL	Local application program output segment size limit allowed in the message queues for each GU call.
LSTT	STATUS	LCL	Local transaction status.

Return, Reason, and Completion Codes for QUERY TRAN

The return and reason codes that can be returned as a result of the QUERY TRAN command are standard for all commands entered through the OM API. Refer to the OM Return and Reason code section in the *IMS Version 9: Common Service Layer Guide and Reference* for the list of codes and their meanings.

Table 127 on page 528 contains the return and reason codes that can be returned to OM from a QUERY TRAN command.

Table 127. Return and Reason Codes for the QUERY TRAN Command

Return Code	Reason Code	Meaning
X'00000000'	X'00000000'	The QUERY TRANcommand completed successfully.
X'00000004'	X'00001000'	The QUERY TRANcommand was not processed on the IMS system as the IMS system is not the command master. No resource information is returned.
X'00000004'	X'00001004'	The QUERY TRAN command was processed for a few resources and a partial list of resources is returned. The command terminated as the resource count to be returned exceeded the maximum number of resources that can be returned by a QUERY TRAN command. The maximum number of resources that can be returned by a QUERY TRAN command is 5000. Re-issue the command with a generic mask or other filters so the number of resources to be returned is less than 5000.
X'00000004'	X'00001010'	No resources were found to be returned. The resource name(s) specified may be invalid or there were no resources that match the filter specified.
X'00000008'	X'00002040'	The QUERY TRAN command has more than one filter value specified or an invalid filter value is specified for the QCNT filter. Only one of the CLASS, STATUS or QCNT filters may be specified. A value of 0 may have been specified for QCNT with LE, GE or, EQ. Or a value of 1 may have been specified for QCNT with LT.
X'00000008'	X'0000204C'	The CLASS value specified is invalid. Confirm the correct CLASS value is specified on the command.
X'0000000C'	X'00003000'	The QUERY TRAN command is successful for at least one resource name. The QUERY TRAN command is not successful for one or more resource names. The Completion Code indicates the reason for the error with the resource name. The Completion Codes that can be returned by the QUERY TRAN command are listed in the QUERY TRAN completion code table.
X'0000000C'	X'00003004'	The QUERY TRAN command is not successful for all the resource name(s) specified. The Completion Code indicates the reason for the error with the resource name. The Completion Codes that can be returned by the QUERY TRAN command are listed in the QUERY TRAN completion code table.
X'00000010'	X'00004014'	The QUERY TRAN command processing terminated as the TRAN keyword is not valid on the RSR tracker.
X'00000010'	X'00004018'	The QUERY TRAN command processing terminated as the RM resource structure is not available.
X'00000010'	X'00004100'	The QUERY TRAN command processing terminated as the resource structure is full.
X'00000010'	X'00004104'	The QUERY TRAN command processing terminated as there is no RM address space.
X'00000010'	X'00004108'	The QUERY TRAN command processing terminated as there is no SCI address space.
X'00000014'	X'00005004'	The QUERY TRAN command processing terminated as a DFSOCMD response buffer could not be obtained.

Table 127. Return and Reason Codes for the QUERY TRAN Command (continued)

Return Code	Reason Code	Meaning
X'00000014'	X'00005008'	The QUERY TRAN command processing terminated as the DFSPOOL storage could not be obtained.
X'00000014'	X'00005100'	The QUERY TRAN command processing terminated due to a RM error.
X'00000014'	X'00005108'	The QUERY TRAN command processing terminated due to a SCI error.

Errors unique to the processing of this command are returned as a completion codes. A completion code is returned for each action against an individual resource.

Table 128 contains completion codes that can be returned on a QUERY TRAN command.

Table 128. Completion Codes for the QUERY TRAN Command

Completion Code	Meaning
0	The QUERY TRAN command completed successfully for the resource.
10	The resource name is unknown to the client that is processing the request. The resource name may have been typed in error or the resource may not be active at this time. If this is a wildcard request there were no matches for the name. Confirm the correct spelling of the resource name is specified on the command.
50	The QUERY TRAN command could not be completed for the resource as CQS is not available. Make sure CQS is available before re-issuing the command.
51	The QUERY TRAN command could not be completed for the resource as there is no resource structure or it is unavailable.
52	The QUERY TRAN command could not be completed for the resource as the resource structure is full.
90	The QUERY TRAN command could not be completed for the resource due to an IMS internal error.
94	The QUERY TRAN command could not be completed for the resource due to a RM error.
98	The QUERY TRAN command could not be completed for the resource due to a CQS error.

Examples for QUERY TRAN Command

This section provides OM API and TSO SPOC input and output examples and explanations for the QUERY command.

Example 1 for QUERY TRAN Command

TSO SPOC input:

```
QRY TRAN NAME(OLCFT117,OLCTMSA*,OLCTB14*) SHOW(QCNT,CLASS,STATUS)
```

TSO SPOC output:

```
Trancode MbrName    CC LCls    LQCnt LclStat
OLCFT117 SYS3        0    1        0 FPE,RESP
OLCTB140 SYS3        0    1        0 BAL
OLCTB141 SYS3        0    5        0
OLCTB148 SYS3        0    1        0 CONV
OLCTMSA1 SYS3        0    1        0 RMT
```

OM API input:

```
CMD(QRY TRAN NAME(OLCFT117,OLCTMSA*,OLCTB14*) SHOW(QCNT,CLASS,STATUS))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2002.163 17:00:12.393828</statime>
<stotime>2002.163 17:00:12.395024</stotime>
<staseq>B7C4BDCC96164DAB</staseq>
<stoseq>B7C4BDCC96610D2B</stoseq>
<rqsttkn1>USRT002 10100012</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<master>SYS3    </master>
<userid>USRT002 </userid>
<verb>QRY </verb>
<kwd>TRAN            </kwd>
<input>QRY TRAN NAME(OLCFT117,OLCTMSA*,OLCTB14*)
 SHOW(QCNT,CLASS,STATUS)</input>
</cmd>
<cmdrsphdr>
<hdr slbl="TRAN" llbl="Trancode" scope="LCL" sort="a" key="1" scroll="no" len="8"
dtype=" CHAR" align="left" />
<hdr slbl="MBR"  llbl="MbrName"  scope="LCL" sort="a" key="4" scroll="no" len="8"
dtype="CHAR" align="left" />
<hdr slbl="CC"   llbl="CC"       scope="LCL" sort="n" key="0" scroll="yes" len="4"
dtype="INT" align="right" />
<hdr slbl="LCLS" llbl="LCls"     scope="LCL" sort="n" key="0" scroll="yes" len="3"
dtype="INT" align="right" />
<hdr slbl="LQ"   llbl="LQCnt"    scope="LCL" sort="d" key="3" scroll="yes" len="8"
dtype="INT" align="right" />
<hdr slbl="LSTT" llbl="LclStat"  scope="LCL" sort="n" key="0" scroll="yes" len="*"
dtype="CHAR" align="left" />
</cmdrsphdr>
<cmdrspdata>
<rsp>TRAN(OLCFT117) MBR(SYS3    ) CC(   0) LCLS(  1) LQ(    0) LSTT(FPE,RESP) </rsp>
<rsp>TRAN(OLCTMSA1) MBR(SYS3    ) CC(   0) LCLS(  1) LQ(    0) LSTT(RMT) </rsp>
<rsp>TRAN(OLCTB140) MBR(SYS3    ) CC(   0) LCLS(  1) LQ(    0) LSTT(BAL) </rsp>
<rsp>TRAN(OLCTB141) MBR(SYS3    ) CC(   0) LCLS(  5) LQ(    0) </rsp>
<rsp>TRAN(OLCTB148) MBR(SYS3    ) CC(   0) LCLS(  1) LQ(    0)  LSTT(CONV) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: A QUERY TRAN command can be issued to obtain information about transactions on one or more IMSs in the IMSplex. This QUERY TRAN command is issued in an IMSplex with one active IMS SYS3 that is not shared queues enabled. The command returns all transactions that match the name OLCFT117 and wild card names OLCTMSA* and OLCTB14*. The output returned is based on the SHOW keyword options. The IMS SYS3 returns the local queue count, local class, and local status for each transaction found that match the NAME parameters. Because the IMS is not shared queues enabled, no global queue count response line is returned.

Example 2 for QUERY TRAN Command

TSO SPOC input:

```
QRY TRAN NAME(PART*) SHOW(QCNT,STATUS,CLASS,PSB)
```

TSO SPOC output:

```
Trancode MbrName    CC PSBname      QCnt LCls    LQCnt LclStat
PART     IMS3        0                 1
PART     IMS2        0 DFSSAM02             4        0
PART     IMS3        0 DFSSAM02             4        0
PART     SYS3        0 DFSSAM02             4        0
PARTROOT IMS3        0                 0
PARTROOT IMS2        0 PARTAPP              1        0
PARTROOT IMS3        0 TPARTAPP             1        0
PARTROOT SYS3        0 TPARTAPP             1        0
```

OM API input:

```
CMD ( QRY TRAN NAME(PART*) SHOW(QCNT,STATUS,CLASS,PSB)    )
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2002.174 23:20:59.381371</statime>
<stotime>2002.174 23:20:59.532136</stotime>
<staseq>B7D2E76D3DE7BD2C</staseq>
<stoseq>B7D2E76D62B68A49</stoseq>
<rqsttkn1>USRT005 10162059</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<master>IMS3  </master>
<userid>USRT005 </userid>
<verb>QRY </verb>
<kwd>TRAN </kwd>
<input>QRY TRAN NAME(PART*) SHOW(QCNT,STATUS,CLASS,PSB)    </input>
</cmd>
<cmdrsphdr>
<hdr slbl="TRAN" llbl="Trancode" scope="LCL" sort="a" key="1" scroll="no"  len="8"
dtype=" CHAR" align="left" />
<hdr slbl="MBR"  llbl="MbrName"  scope="LCL" sort="a" key="4" scroll="no"  len="8"
dtype="CHAR" align="left" />
<hdr slbl="CC"   llbl="CC"       scope="LCL" sort="n" key="0" scroll="yes" len="4"
dtype="INT" align="right" />
<hdr slbl="PSB"  llbl="PSBname"  scope="LCL" sort="n" key="0" scroll="yes" len="8"
dtype="CHAR" align="left" />
<hdr slbl="Q"    llbl="QCnt"     scope="GBL" sort="d" key="2" scroll="yes" len="8"
dtype="INT" align="right" />
<hdr slbl="LCLS" llbl="LCls"     scope="LCL" sort="n" key="0" scroll="yes" len="3"
dtype="INT" align="right" />
<hdr slbl="LQ"   llbl="LQCnt"    scope="LCL" sort="d" key="3" scroll="yes" len="8"
dtype="INT" align="right" />
<hdr slbl="LSTT" llbl="LclStat"  scope="LCL" sort="n" key="0" scroll="yes" len="*"
dtype="CHAR" align="left" />
</cmdrsphdr>
<cmdrspdata>
<rsp>TRAN(PARTROOT) MBR(IMS3    ) CC(   0) Q(    0) </rsp>
<rsp>TRAN(PARTROOT) MBR(IMS3    ) CC(   0) PSB(TPARTAPP) LCLS(  1) LQ(  0) </rsp>
<rsp>TRAN(PART    ) MBR(IMS3    ) CC(   0) Q(    1) </rsp>
<rsp>TRAN(PART    ) MBR(IMS3    ) CC(   0) PSB(DFSSAM02) LCLS(  4) LQ(  0) </rsp>
<rsp>TRAN(PARTROOT) MBR(SYS3    ) CC(   0) PSB(TPARTAPP) LCLS(  1) LQ(  0) </rsp>
<rsp>TRAN(PART    ) MBR(SYS3    ) CC(   0) PSB(DFSSAM02) LCLS(  4) LQ(  0) </rsp>
<rsp>TRAN(PARTROOT) MBR(IMS2    ) CC(   0) PSB(PARTAPP ) LCLS(  1) LQ(  0) </rsp>
<rsp>TRAN(PART    ) MBR(IMS2    ) CC(   0) PSB(DFSSAM02) LCLS(  4) LQ(  0) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: The QUERY TRAN command returns information on the transactions that match the name parameter. This command is issued in an IMSplex that has 3 IMSs - IMS2, IMS3 and SYS3. The resource manager (RM) is using a resource structure and all IMSs are shared queues enabled. The command returns the transactions that match the wildcard name PART*. The output is based on the SHOW options specified. Each IMS processes the command while local information is requested with the SHOW keyword. Each IMS returns its local information for the PSB name, Class, local queue count, and local status associated with the transactions it found that matched the NAME PART*. The master IMS, IMS3, returned the global queue count information from shared queues for all transactions that match the NAME parameter PART*.

Example 3 for QUERY TRAN Command

TSO SPOC input:

```
QRY TRAN QCNT(GT,0) SHOW(AFFIN)
```

TSO SPOC output:

```
Trancode MbrName    CC     QCnt Affinity
PART     IMS3        0        2
SMQ6     IMS3        0        1 SYS3
```

OM API input:

```
CMD ( QRY TRAN QCNT(GT,0) SHOW(AFFIN))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2002.174 23:39:11.140576</statime>
<stotime>2002.174 23:39:11.191856</stotime>
<staseq>B7D2EB7E6CAE0127</staseq>
<stoseq>B7D2EB7E79330E8D</stoseq>
<rqsttkn1>USRT005 10163911</rqsttkn1>
<rc>0200000C</rc>
<rsn>00003000</rsn>
</ctl>
<cmderr>
<mbr name="SYS3    ">
<typ>IMS     </typ>
<styp>DBDC    </styp>
<rc>00000004</rc>
<rsn>00001000</rsn>
</mbr>
<mbr name="IMS2    ">
<typ>IMS     </typ>
<styp>DCCTL   </styp>
<rc>00000004</rc>
<rsn>00001000</rsn>
</mbr>
</cmderr>
<cmd>
<master>IMS3    </master>
<userid>USRT005 </userid>
<verb>QRY </verb>
<kwd>TRAN </kwd>
<input>QRY TRAN QCNT(GT,0) SHOW(AFFIN)  </input>
</cmd>
<cmdrsphdr>
<hdr slbl="TRAN" llbl="Trancode" scope="LCL" sort="a" key="1" scroll="no" len="8"
dtype=" CHAR" align="left" />
```

```
<hdr slbl="MBR"  llbl="MbrName"  scope="LCL" sort="a" key="4" scroll="no" len="8"
dtype="CHAR" align="left" />
<hdr slbl="CC"    llbl="CC"      scope="LCL" sort="n" key="0" scroll="yes" len="4"
dtype="INT" align="right" />
<hdr slbl="Q"     llbl="QCnt"    scope="GBL" sort="d" key="2" scroll="yes" len="8"
dtype="INT" align="right" />
<hdr slbl="AFIN" llbl="Affinity" scope="GBL" sort="n" key="0" scroll="yes" len="7"
dtype="CHAR" align="left" />
</cmdrsphdr>
<cmdrspdata>
<rsp>TRAN(PART    ) MBR(IMS3    ) CC(   0) Q(    2) AFIN(       ) </rsp>
<rsp>TRAN(SMQ6    ) MBR(IMS3    ) CC(   0) Q(    1) AFIN(SYS3   ) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: The `QUERY TRAN QCNT(GT,0)` command returns all transactions that have a queue count greater than 0. This command is issued in an IMSplex with 3 IMSs - IMS2, IMS3 and SYS3. RM is using the resource structure and all IMSs are shared queues enabled. In a shared queues environment, the `QUERY TRAN QCNT(GT,0)` command is only processed by the master IMS because it can obtain the global queue counts. All transactions that have a queue count greater than 0 are returned by the master IMS which is IMS3. When SHOW(AFFIN) is specified, any affinity of the messages to an IMS is returned. In this example, transactions SMQ6 has a message queued that can be processed only on the IMS SYS3.

Example 4 for QUERY TRAN Command

TSO SPOC input:

```
QRY TRAN NAME(FPTR*) STATUS(FPE,RESP)
```

TSO SPOC output:

```
Trancode MbrName    CC LclStat
FPTRN01  IMS3        0 RESP
FPTRN01  SYS3        0 RESP
FPTRN02  IMS3        0 FPE,RESP
FPTRN02  SYS3        0 FPE,RESP
FPTRN03  IMS3        0 FPE,RESP
FPTRN03  SYS3        0 FPE,RESP
FPTRN04  IMS3        0 FPE,RESP
FPTRN04  SYS3        0 FPE,RESP
FPTRN05  IMS3        0 FPE,RESP
FPTRN05  SYS3        0 FPE,RESP
FPTRN06  IMS3        0 FPE,RESP
FPTRN06  SYS3        0 FPE,RESP
```

OM API input:

```
CMD ( QRY TRAN NAME(FPTR*) STATUS(FPE,RESP)  )
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2002.174 23:47:42.864254</statime>
<stotime>2002.174 23:47:42.865685</stotime>
<staseq>B7D2ED667137E629</staseq>
<stoseq>B7D2ED6671915729</stoseq>
<rqsttkn1>USRT005 10164742</rqsttkn1>
<rc>0200000C</rc>
<rsn>00003000</rsn>
</ctl>
<cmderr>
```

```
<mbr name="IMS2    ">
<typ>IMS     </typ>
<styp>DCCTL   </styp>
<rc>00000008</rc>
<rsn>0000200C</rsn>
</mbr>
</cmderr>
<cmd>
<master>IMS3    </master>
<userid>USRT005 </userid>
<verb>QRY </verb>
<kwd>TRAN  </kwd>
<input>QRY TRAN NAME(FPTR*) STATUS(FPE,RESP)  </input>
</cmd>
<cmdrsphdr>
<hdr slbl="TRAN" llbl="Trancode" scope="LCL" sort="a" key="1" scroll="no" len="8"
dtype=" CHAR" align="left" />
<hdr slbl="MBR"  llbl="MbrName"  scope="LCL" sort="a" key="4" scroll="no" len="8"
dtype="CHAR" align="left" />
<hdr slbl="CC"   llbl="CC"       scope="LCL" sort="n" key="0" scroll="yes" len="4"
dtype="INT" align="right" />
<hdr slbl="LSTT" llbl="LclStat"  scope="LCL" sort="n" key="0" scroll="yes" len="*"
dtype="CHAR" align="left" />
</cmdrsphdr>
<cmdrspdata>
<rsp>TRAN(FPTRN01 ) MBR(IMS3    ) CC(   0) LSTT(RESP) </rsp>
<rsp>TRAN(FPTRN03 ) MBR(IMS3    ) CC(   0) LSTT(FPE,RESP) </rsp>
<rsp>TRAN(FPTRN02 ) MBR(IMS3    ) CC(   0) LSTT(FPE,RESP) </rsp>
<rsp>TRAN(FPTRN06 ) MBR(IMS3    ) CC(   0) LSTT(FPE,RESP) </rsp>
<rsp>TRAN(FPTRN04 ) MBR(IMS3    ) CC(   0) LSTT(FPE,RESP) </rsp>
<rsp>TRAN(FPTRN05 ) MBR(IMS3    ) CC(   0) LSTT(FPE,RESP) </rsp>
<rsp>TRAN(FPTRN01 ) MBR(SYS3    ) CC(   0) LSTT(RESP) </rsp>
<rsp>TRAN(FPTRN03 ) MBR(SYS3    ) CC(   0) LSTT(FPE,RESP) </rsp>
<rsp>TRAN(FPTRN02 ) MBR(SYS3    ) CC(   0) LSTT(FPE,RESP) </rsp>
<rsp>TRAN(FPTRN06 ) MBR(SYS3    ) CC(   0) LSTT(FPE,RESP) </rsp>
<rsp>TRAN(FPTRN04 ) MBR(SYS3    ) CC(   0) LSTT(FPE,RESP) </rsp>
<rsp>TRAN(FPTRN05 ) MBR(SYS3    ) CC(   0) LSTT(FPE,RESP) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: A `QUERY TRAN NAME(FPTR*)STATUS(FPE,RESP)` is issued to obtain all transactions that match the wildcard name FPTR* and also have one or more of the following status conditions:

- FPE - Fast Path enabled
- RESP - Response mode

This command is issued in an IMSplex with 3 IMSs - IMS2, IMS3 and SYS3. RM is using the resource structure and all IMSs are shared queues enabled. All IMSs process the command because the status is local. Each IMS returns the transactions that are defined locally that match the wildcard name FPTR* and have the status of FP, RESP, or both. The status condition that resulted in the transaction name being returned is also returned even though SHOW(STATUS) is not specified.

Example 5 for QUERY TRAN Command

TSO SPOC input:

```
QRY TRAN CLASS(6)
```

TSO SPOC output:

```
Trancode MbrName    CC LCls
HPCSTCL6 IMS3        0    6
HPCSTCL6 SYS3        0    6
OLCTB241 IMS2        0    6
OLCTB241 IMS3        0    6
```

```
OLCTB241 SYS3        0    6
SKS6     IMS2        0    6
SKS6     IMS3        0    6
SKS6     SYS3        0    6
```

OM API input:

```
CMD ( QRY TRAN CLASS(6)   )
```

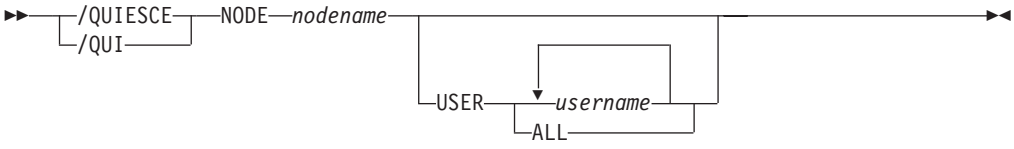
OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2002.174 23:54:03.498534</statime>
<stotime>2002.174 23:54:03.500563</stotime>
<staseq>B7D2EED171826C43</staseq>
<stoseq>B7D2EED172013A43</stoseq>
<rqsttkn1>USRT005 10165403</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<master>IMS3    </master>
<userid>USRT005 </userid>
<verb>QRY </verb>
<kwd>TRAN            </kwd>
<input>QRY TRAN CLASS(6)  </input>
</cmd>
<cmdrsphdr>
<hdr slbl="TRAN" llbl="Trancode" scope="LCL" sort="a" key="1" scroll="no" len="8"
dtype=" CHAR" align="left" />
<hdr slbl="MBR"  llbl="MbrName"  scope="LCL" sort="a" key="4" scroll="no" len="8"
dtype="CHAR" align="left" />
<hdr slbl="CC"   llbl="CC"       scope="LCL" sort="n" key="0" scroll="yes" len="4"
dtype="INT" align="right" />
<hdr slbl="LCLS" llbl="LCls"     scope="LCL" sort="n" key="0" scroll="yes" len="3"
dtype="INT" align="right" />
</cmdrsphdr>
<cmdrspdata>
<rsp>TRAN(SKS6    ) MBR(IMS3    ) CC(   0) LCLS(  6) </rsp>
<rsp>TRAN(HPCSTCL6) MBR(IMS3    ) CC(   0) LCLS(  6) </rsp>
<rsp>TRAN(OLCTB241) MBR(IMS3    ) CC(   0) LCLS(  6) </rsp>
<rsp>TRAN(SKS6    ) MBR(SYS3    ) CC(   0) LCLS(  6) </rsp>
<rsp>TRAN(HPCSTCL6) MBR(SYS3    ) CC(   0) LCLS(  6) </rsp>
<rsp>TRAN(OLCTB241) MBR(SYS3    ) CC(   0) LCLS(  6) </rsp>
<rsp>TRAN(SKS6    ) MBR(IMS2    ) CC(   0) LCLS(  6) </rsp>
<rsp>TRAN(OLCTB241) MBR(IMS2    ) CC(   0) LCLS(  6) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: A `QUERY TRAN CLASS(6)` is issued to obtain all transactions that have a local class value of 6. This command is issued in an IMSplex with 3 IMSs - IMS2, IMS3 and SYS3. RM is using the resource structure and all IMSs are shared queues enabled. All the IMSs that process the command as CLASS, are local. The NAME parameter is not specified so each IMS returns all the transactions defined locally with a class value 6. The class value that resulted in the transaction name being returned is also returned even though SHOW(CLASS) is not specified.

Chapter 43. /QUIESCE

Format



Environments and Keywords

Table 129 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 129. Valid Environments for the /QUIESCE Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/QUIESCE	X		X
NODE	X		X
USER	X		X

Usage

/QUIESCE initiates the shutdown and deallocates the user for the specified ISC node. The /QUIESCE NODE command is valid for ISC nodes only.

/QUIESCE resets preset mode, test mode, response mode, lock node, lock lterm, pstop lterm, and purge lterm because these statuses are not significant and are not kept after a logon or restart./QUIESCE also takes other actions depending on the recovery settings for the node:

RCVYCONV=NO
/QUIESCE causes any IMS conversations (active and held) to be terminated. Any conversational message that is queued or being processed has its output response message delivered asynchronously.

RCVYFP=NO
/QUIESCE causes Fast Path status and messages to be discarded

If global resource information is not kept in Resource Manager, /QUIESCE deallocates the user and resets status locally. If global resource information is kept in Resource Manager, /QUIESCE deallocates the user and resets status globally. If the user has no significant status, /QUIESCE deletes the user in Resource Manager. If the node has no significant status, and there are no other half-sessions for the node, /QUIESCE deletes the node in Resource Manager.

If ROUTE is specified, it should be specified with ROUTE(*). The command fails if not routed to the IMS where the node is active.

NODE
Specifies the VTAM node for the user to be shut down and deallocated.

© Copyright IBM Corp. 1974, 2003

USER
If the USER keyword is omitted, all half-sessions of an ISC node are affected. The half-sessions must be connected.

Example for /QUIESCE Command

Entry ET:

```
/QUIESCE NODE CAL USER LAX
```

Response ET:

```
/DFS058I  QUIESCE COMMAND COMPLETED
```

Explanation: The half-session of node CAL using user LAX is shut down.

Chapter 44. /RCLSDST

Format

Environments

Table 130 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command can be issued.

Table 130. Valid Environments for the /RCLSDST Command

Command	DB/DC	DBCTL	DCCTL
/RCLSDST	X		X

Usage

/RCLSDST causes IMS to disconnect the VTAM terminal from which the command is entered. If you are in an active conversational mode, /EXIT or /HOLD must be entered before /RCLSDST is executed. If this command is issued by a signed on user, the user is signed off.

This command does not reset preset mode.

/RCLSDST resets preset mode, response mode, test mode, lock node, lock lterm, pstop lterm, and purge lterm because these statuses are not significant and, therefore, are not kept after logons and restart. /RCLSDST also takes other actions depending on the recovery settings for the node:

RCVYSTSN=NO
: /RCLSDST acts like a /CHANGE NODE COLDSESS command for FINANCE and SLUP nodes by setting the session status to 'cold'. /RCLSDST acts like a /QUIESCE NODE command for ISC (LU6.1) nodes by initiating the shutdown and deallocating the user for the specified node. This action changes the session status to 'cold'. With these actions taken by the /RCLSDST command, the next session initiation request for this node is allowed to again attempt a session cold start. For ETO nodes, the control block structure could be deleted, if no significant status exists.

RCVYCONV=NO
: /RCLSDST causes any held IMS conversations to be terminated. Any conversational message that is queued or being processed has its output response message delivered asynchronously.

RCVYFP=NO
: /RCLSDST causes Fast Path status and messages to be discarded.

If global resource information is not kept in Resource Manager, /RCLSDST logs a node off and resets status locally. If global resource information is kept in Resource Manager, /RCLSDST logs a node off and resets status globally. If the node has no status, /RCLSDST deletes the node in Resource Manager.

© Copyright IBM Corp. 1974, 2003

Example for /RCLSDST Command

Entry ET:

```
/RCLSDST
```

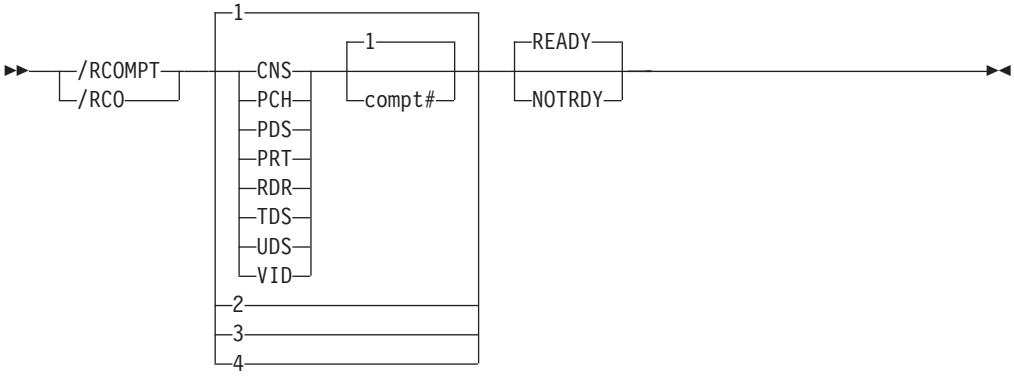
Response ET:

```
DFS058I  RCLSDST COMMAND COMPLETED
```

Explanation: The entering terminal is logged off IMS.

Chapter 45. /RCOMPT

Format



Environments and Keywords

Table 131 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 131. Valid Environments for the /RCOMPT Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/RCOMPT	X		X
CNS	X		X
NOTRDY	X		X
PCH	X		X
PDS	X		X
PRT	X		X
RDR	X		X
READY	X		X
TDS	X		X
UDS	X		X
VID	X		X

Usage

/RCOMPT sets a particular VTAM terminal component to a ready/not ready state. Output messages queued for a particular component will not be sent unless the component is ready. Depending on terminal type, output operations for other components can continue.

Note: Defaults are READY and 1.

The ready/not ready state set by the /RCOMPT command can be altered by the following:

- Another /RCOMPT command

© Copyright IBM Corp. 1974, 2003

- A /COMPT, /START, or /RSTART command
- An I/O error on the terminal component

The command format takes one of the following forms:

- A keyword is used.

  A search is made of the components (as defined in the TERMINAL macro during IMS system definition or logon descriptor) for the component defined that corresponds to the specified keyword. When a match is found, that component type is made ready or not ready as specified by the command.
- A keyword is used with a number other than 1 following the keyword.

  The corresponding occurrence of that component type is made ready or not ready, as specified by the command.
- Number 1 through 4 is used instead of a keyword.

  The component affected is the one defined in that position during system definition or logon descriptor independent of component type. For more information about component support see Chapter 5, "/ASSIGN," on page 99 For more information on the keywords, see "Keywords" on page 17.

When using ISC, only parameters 1, 2, 3, and 4 are valid.

Example for /RCOMPT Command

Entry ET:

```
/RCOMPT VID 2 READY
```

Response ET:

```
DFS058I  RCOMPT COMMAND COMPLETED
```

Explanation: The second display component is declared operable to IMS.

Entry ET:

```
/RCOMPT 4 READY
```

Response ET:

```
DFS058I  RCOMPT COMMAND COMPLETED
```

Explanation: The fourth component defined is declared ready to IMS.

Chapter 46. /RDISPLAY

Format

Environments

Table 132 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keyword can be issued.

Table 132. Valid Environments for the /RDISPLAY Command and Keyword

Command/Keyword	DB/DC	DBCTL	DCCTL
/RDISPLAY	X		X
MASTER	X		X

Usage

/RDISPLAY references the terminal assigned as the master terminal and displays either:

- The logical terminal name and the line and physical terminal numbers
- The logical terminal name and the VTAM NODE name

If the 3270 master terminal capability was generated during IMS system definition, the logical terminal name, line, and physical terminal number of the secondary master terminal are also displayed.

MASTER
Specifies the identity of the terminal designated as the master terminal.

Examples

Example 1 for /RDISPLAY Command

Entry ET:

```
/RDISPLAY MASTER
```

Response ET:

```
LTERM CNTRL
PTERM 3-1
*91010/123704*
```

Explanation: CNTRL is the master terminal logical terminal and is assigned to LINE 3 PTERM 1.

Example 2 for /RDISPLAY Command

Entry ET:

```
/RDISPLAY MASTER
```

Response ET:

© Copyright IBM Corp. 1974, 2003

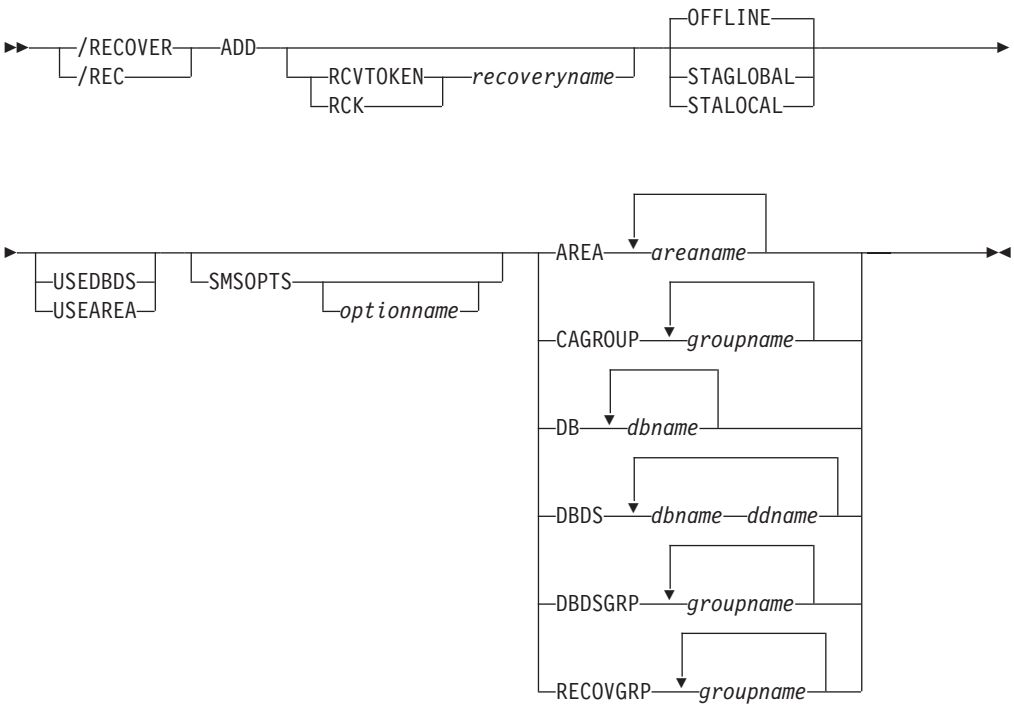
```
LTERM CTRL1
PTERM 4-2
LTERM CTRL2
PTERM 4-4
*91010/12370*
```

Explanation: CTRL1 is the primary master terminal logical terminal and is assigned to LINE 4 PTERM 2. CTRL2 is the secondary master terminal logical terminal and is assigned to LINE 4 PTERM 4.

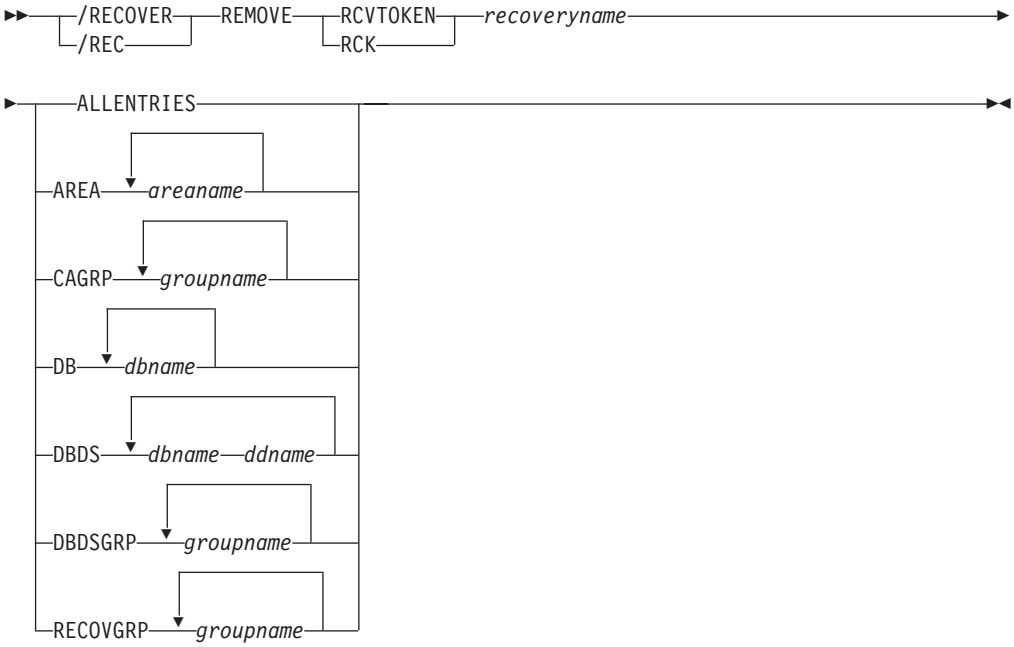
Chapter 47. /RECOVER

Format

/RECOVER Command: ADD

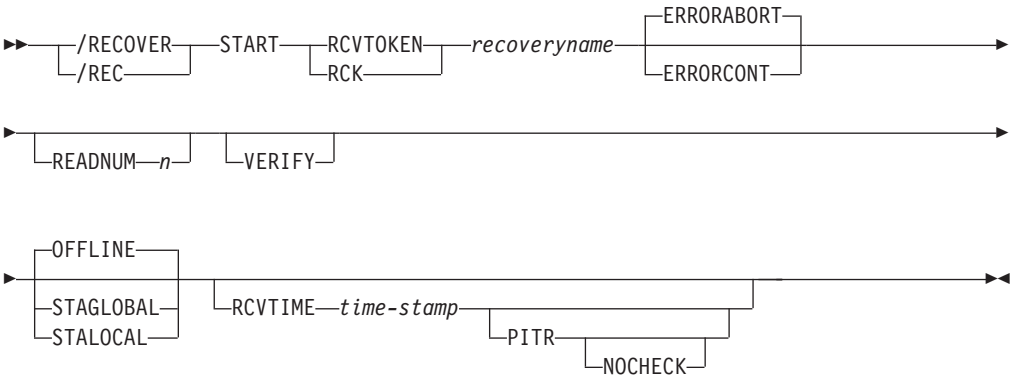


/RECOVER Command: REMOVE

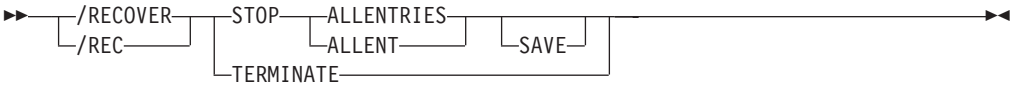


© Copyright IBM Corp. 1974, 2003

/RECOVER Command: START



/RECOVER Command: STOP through TERMINATE



Environments and Keywords

Table 133 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 133. Valid Environments for the /RECOVER Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/RECOVER	X	X	
ADD	X	X	
ALLENTRIES	X	X	
AREA	X	X	
CAGROUP	X	X	
DB	X	X	
DBDS	X	X	
DBDSGRP	X	X	
ERRORABORT	X	X	
ERRORCONT	X	X	
NOCHECK	X	X	
OFFLINE	X	X	
PITR	X	X	
RCVTIME	X	X	
RCVTOKEN	X	X	
READNUM	X	X	
RECOVGRP	X	X	
REMOVE	X	X	
SMSOPTS	X	X	

Table 133. Valid Environments for the /RECOVER Command and Keywords (continued)

Command / Keywords	DB/DC	DBCTL	DCCTL
STAGLOBAL	X	X	
STALOCAL	X	X	
START	X	X	
STOP	X	X	
USEAREA	X	X	
USEDBDS	X	X	
VERIFY	X	X	

Usage

/RECOVER commands are used with the recovery list of database data sets and areas.

In an IMSplex, the /RECOVER command initiates the operation of the Database Recovery facility. OM sends the /RECOVER command to one IMS.

/RECOVER ADD

Usage

The /RECOVER ADD command is used to add database data sets and areas to a list (recovery list) of database data sets and areas to be recovered using the Online Recovery Service. The database data sets and areas can be specified as database data sets, areas, databases, or groups.

Successful completion of a /RECOVER ADD command results in the specified database data sets and areas being added to a recovery list. Database data sets and areas can be added to a recovery list by specifying one or more database data sets, databases, change accumulation groups (CAGROUP), database data set groups (DBDSGRP), or recovery groups (RECOVGRP). If a database or group is specified, all database data sets and areas making up the database or group are added to a recovery list. If the a specified database is a master database for a partitioned HALDB, all database data sets from all partitions that make up the HALDB are added to a recovery list. All groups (including databases) are defined in DBRC.

Database data sets and areas must be registered with DBRC to be recovered with the IMS Recovery Services. If the database data set, area, or group name is not known to DBRC, it is not added to a recovery list and a message is issued.

If a database data set or area specified in a /RECOVER ADD command (individually or as part of a group) is already on a recovery list, processing for the duplicate is ignored and a message is issued. Other database data sets and areas that are not duplicates are processed normally.

This command can be issued in IMS DBCTL and IMS DB/DC environments.

Examples of using the /RECOVER ADD command are shown in:

- “Example 1 for /RECOVER ADD Command” on page 556
- “Example 2 for /RECOVER ADD Command” on page 556

- "Example 3 for /RECOVER ADD Command" on page 556
- "Example 4 for /RECOVER ADD Command" on page 556
- "Example 5 for /RECOVER ADD Command" on page 556
- "Example 6 for /RECOVER ADD Command" on page 557

OFFLINE

With this option, the database remains offline from the IMS system at the completion of the recovery. This option is used to allow the database administrator to verify that the recovery has completed successfully and the databases are ready for use. This is the default if it is not specified on the /RECOVER START command.

The IMS system performing the recovery has EXCLUSIVE authorization of the database to perform the recovery. The OFFLINE option allows the IMS system to unauthorize the databases at the completion of the recovery without issuing a START command. This will leave the databases available for processing. If a PITR option has been used in a recovery, the database data set in the RECON is IMAGE COPY NEEDED.

STAGLOBAL

This option is used when the full function database(s) and Fast Path areas are used in a SYSPLEX data sharing environment. A /START DB command with the GLOBAL option is issued on the IMS system which ran the recovery for all full function databases affected by recovery. A /START AREA command with the GLOBAL option is used on the IMS system which ran the recovery for all Fast Path areas recovered. This option makes the database(s) available to IMS transactions as soon as all the DBDSs in the recovery list have been recovered. Authorization to use the database is returned to IMS.

STALOCAL

This option is used when the DRF is being executed in the operational IMS system where the database(s) is used. A /START DB command is issued on the IMS system which ran the recovery for all full function databases affected by recovery. A /START AREA is issued on the IMS system which ran the recovery for all Fast Path areas recovered. Fast Path databases' are not started with the /START DB command. Only one option needs to be specified and it applies to all DBDSs and areas added to the recovery list.

USEDBDS or USEAREA

These options are for Fast Path areas, full function, and HALDB DBDS. When specified, DRF does not need to restore the image copy before applying log updates. You can use this option when you restore image copies prior to recovery.

Note: You must restore non-standard image copies prior to recovery.

This parameter determines whether or not DRF will restore the image copy for a DBDS before applying log updates.

RCVTOKEN | RCK

Specifies the unique identifier associated with a recovery list that the /RECOVER ADD command operates against. RCVTOKEN is optional. If it is not specified, IMS generates a recovery name. However, if the command is intended to add entries to an existing recovery list, RCVTOKEN must be specified with the intended recovery list token *recoveryname*. When RCVTOKEN is specified in a command, it must come before any keywords that identify the names of database data sets, areas, or groups.

recoveryname
Specifies the unique recovery token associated with the recovery list that the /RECOVER ADD command operates against. This token can be up to eight characters in length.

SMSOPTS
Specifies that the DFSMSdss options are to be associated with the entries being added to the recovery list. The options are only used when restoring image copies created by the Image Copy 2 utility.

optionname
Specifies a unique SMS option.

DELCAT
Specifies that the data set to be restored using the DFSMSdss optional keyword, DELETECATALOGENTRY.

Caution: This option is to be used with extreme care. DELCAT is required if SMSOPTS is supplied. This option allows you to recover from a scenario where an entire volume or volumes are lost but the catalog entries remain. When this option is specified, SMS deletes the prior catalog entries for the database data sets and areas being restored as part of recovery. Refer to the *DFSMSdss Storage Administration Reference* manual before using this option. Read the caution under the DELETECATALOGENTRY option of the RESTORE command.

AREA
Specifies that one or more Fast Path areas are to be added to a recovery list.

areaname
Specifies a unique Fast Path area.

CAGROUP
Specifies that one or more change accumulation groups as defined in the RECON data sets are to have their database data sets and areas added to a recovery list.

groupname
Specifies that the database data sets and areas belonging to the named CA group are to be added to the recovery list.

DB
Specifies that all the areas or full function database data sets for one or more databases are to be added to a recovery list.

dbname
Specifies the database and the associated database data sets or areas that are to be added to a recovery list.

DBDS
Specifies that one or more full function database data sets are to be added to a recovery list.

dbname ddname
Specifies a full function database data set to be added to a recovery list. Full function database data sets are specified with the /RECOVER ADD DBDS command as an ordered pair. The first member of the pair is the database name. The second member is the DD name. If more than one full function database data set is specified, the complete ordered pair must be specified for each database data set. All parameters must be separated by at least one blank space.

DBDSGRP
Specifies that one or more DBDS groups as defined in the RECON data sets will have their database data sets and areas added to a recovery list.

groupname
Specifies that the database data sets and areas belonging to the named DBDS group are to be added to the recovery list.

RECOVGRP
Specifies that the listed groups are recovery groups. A recovery group is a group of full-function databases, DEDB areas, or both that the user defines to IMS as related. All DBDSs that make up the full-function databases and all the DEDB areas making up the recover groups specified in the command are added to a recovery list.

For additional information on recovery groups, see the *IMS Version 9: DBRC Guide and Reference* and the *IMS Version 9: Administration Guide: Database Manager*.

groupname
Specifies the unique name of the group whose database data sets and areas are to be added to a recovery list.

/RECOVER REMOVE

Usage

A /RECOVER REMOVE command removes some or all database data sets and areas from the recovery list. It can only be issued prior to issuing the /RECOVER START command. Use the /RECOVER STOP command to remove entries after recovery has started.

- If /RECOVER REMOVE is issued before the /RECOVER START command, database data sets and areas specified on the /RECOVER REMOVE command individually or as part of databases or groups are removed from the recovery list. A subsequent /RECOVER START command initiates recovery for the remaining members in the recovery list.
- If a /RECOVER REMOVE ALLENTRIES command is issued before the /RECOVER START command, all elements in the list are removed and the recovery list is eliminated.
- If the /RECOVER REMOVE command is issued after the /RECOVER START command, the /RECOVER REMOVE command is rejected.

If /RECOVER REMOVE is issued with one or more databases or groups, all database data sets and areas that are part of the database or group specified are removed from the recovery list. If a /RECOVER REMOVE command results in the removal of every data set or area entry from the recovery list, the recovery list is eliminated.

This command executes in IMS DBCTL and IMS DB/DC environments.

Examples of using the /RECOVER REMOVE command are shown in:

- “Example 1 for /RECOVER REMOVE Command” on page 557
- “Example 2 for /RECOVER REMOVE Command” on page 557
- “Example 3 for /RECOVER REMOVE Command” on page 557
- “Example 4 for /RECOVER REMOVE Command” on page 558

RCVTOKEN | RCK
Specifies the unique identifier associated with the recovery list that the /RECOVER

REMOVE command operates against. The RCVTOKEN keyword must come before any keywords that identify the names of database data sets, areas, or groups.

recoveryname
Specifies the unique recovery token associated with the recovery list that the /RECOVER REMOVE command operates against. This token can be up to eight characters in length.

ALLENTRIES
Specifies that the recovery list is to be eliminated.

AREA
Specifies that one or more Fast Path areas are to be removed from the recovery list.

areaname
Specifies a unique Fast Path area.

CAGROUP
Specifies that the database data sets and areas of one or more change accumulation groups as defined in the RECON data sets are to be removed from the recovery list.

groupname
Specifies that the database data sets and areas belonging to a specific CA group are to be removed from the recovery list.

DATAGROUP
Specifies that the database data sets and areas of one or more database groups (as defined in the RECONs) are to be removed from the recovery list.

groupname
Specifies the database data sets and areas of the unique group name that are to be removed from the recovery list.

DB
Specifies that the full function database data sets or Fast Path areas making up one or more databases are to be removed from the recovery list.

dbname
Specifies database data sets or areas of the database that are to be added to a recovery list.

DBDS
Specifies that one or more full function database data sets are to be removed from the recovery list.

dbname
Specifies the database data sets or areas of the database that are to be removed from the recovery list.

ddname
Specifies the DD name of the database data set. If DBDS is specified on the /RECOVER REMOVE command, *dbname* and *ddname* must be specified together.

DBDSGRP
Specifies that the database data sets and areas of one or more DBDS groups as defined in the RECON data sets are to be removed from the recovery list.

groupname
Specifies the database data sets and areas of the group that are to be removed from the recovery list.

RECOVGRP
Specifies that this group is a recovery group. All DBDSs that make up the full-function databases and all the DEDB areas are removed from the recovery list.

groupname
Specifies the database data sets and areas of the database that are to be removed from the recovery list.

/RECOVER START

Usage

Use the /RECOVER START command to start the recovery process for all the members of a recovery list, which includes performing the following tasks:

- Image copies are restored to the database data sets and areas in the recovery list.
- Change accumulation data is applied to the database data sets and areas in the recovery list.
- The database data sets and areas are brought up to date by applying data changes from log data sets (or up to the recovery time).
- Log data sets cached to a VTS are staged to DASD as a user option.

Only one /RECOVER START command is allowed to execute in one IMS at a time. If DRF is to run in conjunction with multiple IMSs simultaneously, ensure that log contention situations do not occur. To avoid log contention situations, ensure that recovery instances that would read the same log data sets do not execute simultaneously in multiple IMSs.

You can choose to automatically /START any or all members of the recovery list after successful completion of recovery, either on all IMS systems on which they are defined or just the one where the recovery is executed.

If ERRORABORT is in effect, the recovery list will not start until all the DBDSs in the list can be authorized for recovery. This is not true if the ERRORCONT parameter is specified on the /RECOVER START command. Instead, the recovery will continue.

If coordinated online change removes from the system any database data set or area that has been previously added to a recovery list, message DFS4266I with reason code NOT FOUND will be issued after the /RECOVER START command for that recovery list is entered.

Examples of using the /RECOVER START command are shown in:

- “Example 1 for /RECOVER START Command” on page 558
- “Example 2 for /RECOVER START Command” on page 558
- “Example 3 for /RECOVER START Command” on page 559
- “Example 4 for /RECOVER START Command” on page 559
- “Example 5 for /RECOVER START Command” on page 559

RCVTOKEN | RCK
Specifies the token of the recovery list that will be processed. When

RCVTOKEN is specified in a command, it must come before any keywords that identify the names of database data sets, areas, or groups.

recoveryname
Specifies the unique recovery token associated with the recovery list to be processed. This token can be up to eight characters in length.

ERRORABORT
Specifies that recovery stops for all entries in the recovery list if recovery of any database data set can not be completed. ERRORABORT is the default.

ERRORCONT
Specifies that recovery is to continue if recovery is able to complete processing for any database data set.

READNUM *n*
Specifies the number of input devices used in parallel during recovery. Image copies are restored using the number of input devices specified by *n*. When image copies are restored, log data sets are read using the number of input devices specified by *n*.

VERIFY
Specifies the report only execution and obtains a list of the log, change accumulation, and image copy data sets required to process recovery for the associated recovery list. This option allows users to perform set up procedures before invoking the recovery process.

OFFLINE
This option leaves all the databases offline after the recovery is complete. When the recovery is complete, DRF will unauthorize the databases. This leaves the databases in a state which allows normal DBRC processing.

STAGLOBAL
This option is used in a sysplex data sharing environment or with two IMSs sharing data on the same central processing complex (CPC). A `/START DB` command with the GLOBAL option is issued internally for DL/I databases and a `/START AREA` command with the GLOBAL option is issued for the Fast Path areas. OFFLINE, STALOCAL, or STAGLOBAL options specified on the `/RECOVER START` command are used only for database data sets and areas that were added without any of these options specified.

STALOCAL
This option is used to start the databases on the IMS system that ran the DRF recovery. A `/START DB` command with the LOCAL option is issued internally.

RCVTIME
Specifies the time stamp to which a point in time or time stamp recovery is to be performed.

For information about time stamp recovery characteristics, see *IMS Version 9: Utilities Reference: Database and Transaction Manager*.

time-stamp
The time-stamp must have a format that is recognizable to IMS. For additional information on acceptable time stamp formats, see the *IMS Version 9: DBRC Guide and Reference*. Note the UTC offset portion of a time stamp cannot be specified using a symbolic value in this command.

Additionally, the timestamp must by surrounded by single quotation marks (’). For example:

```
/RECOVER START RCVTOKEN R1 RCVTIME '022671213156'
```

PITR
Specifies that a timestamp recovery (TSR) will be performed to the time specified with the RCVTIME parameter regardless if there are any active database allocations for the specified database data sets.

NOCHECK
If a portion of the database data sets making up a database are in the recovery list being started, NOCHECK specifies that the Online Recovery Service will not stop a timestamp recovery or a timestamp recovery to any prior point in time (PITR) if one of the following situations occurs:

- All members of the recovery group are not in the same recovery list.
- All members of the recovery group are not being recovered to an equivalent point in time. For more information, see the *IMS Version 9: DBRC Guide and Reference*, *IMS Version 9: Operations Guide*, and *IMS Version 9: Administration Guide: Database Manager*.

/RECOVER STOP

Usage

A /RECOVER STOP command stops recovery for all database data sets and areas on the recovery list. The command can only be issued for a recovery list that has had /RECOVER START issued against it. If the /RECOVER STOP is issued before the /RECOVER START command, it is rejected. If it is issued after the /RECOVER START command, recovery is stopped for all database data sets in the recovery list. After a /RECOVER STOP command successfully processes, subsequent /RECOVER STOP commands are rejected.

If /RECOVER STOP ALLENT is issued, all recovery processing for the affected recovery list halts, and the existing recovery list is deleted.

Examples of using the /RECOVER STOP command are shown in:

- "Example 1 for /RECOVER STOP" on page 560
- "Example 2 for /RECOVER STOP" on page 560
- "Example 3 for /RECOVER STOP" on page 560

ALLENTRIES
Specifies that recovery is to be aborted for all database data sets and areas (all entries) in the recovery list.

SAVE
Specifies that the recovery list is not to be deleted when recovery is stopped. This parameter is only allowed with the ALLENT parameter after recovery has been initiated with the /RECOVER START command.

AREA
Specifies that recovery processing is to be stopped for the specified Fast Path areas.

areaname
Specifies a unique Fast Path area.

CAGROUP
Specifies that recovery processing is to be stopped for the database data sets and areas making up the specified change accumulation groups as defined in the RECON data sets.

groupname
Specifies the unique name of the group whose database data sets and areas are to have recovery processing be stopped.

DATAGROUP
Specifies that recovery processing is to be stopped for database data sets and areas making up the specified database groups as defined in the RECON data sets.

groupname
Specifies the unique name of the group whose database data sets and areas are to have recovery processing be stopped.

DB
Specifies that recovery processing is to be stopped for the full function database data sets or Fast Path areas that make up the specified databases.

dbname
Specifies the database whose database data sets or areas are to be added to a recovery list.

DBDS
Specifies that recovery processing is to be stopped for the specified full function database data sets.

dbname
Specifies the database whose database data sets or areas are to be added to a recovery list.

ddname
The 8 character identifier associated with the data set name and data set characteristics. *dbname* and *ddname* must be specified together if DBDS is specified on the `/RECOVER REMOVE` command.

DBDSGRP
Specifies that recovery processing is to be stopped for the database data sets and areas making up the specified DBDS groups as defined in the RECON data sets.

groupname
Specifies the unique name of the group whose database data sets and areas are to have recovery processing be stopped.

RECOVGRP
Specifies that this group is a recovery group. A recovery group is a group of full-function databases or DEDB areas that are considered to be related. All DBDSs that make up the full-function databases and all the DEDB areas are removed from the recovery list.

groupname
Specifies the unique name of the group whose database data sets and areas are to have recovery processing be stopped.

/RECOVER TERMINATE

Usage

The TERMINATE option will delete all lists in BEING BUILT status and terminate the DRS address space.

If a recovery is in progress the recovery will ignore the `/RECOVER TERMINATE` command and continue to process. When it completes, another `/RECOVER TERMINATE`

command is required. It will not take effect automatically. If you would like to force DRF down while a recovery is running, you must issue the /RECOVER STOP ALLENT command first. This will stop the recovery and then you may enter the /RECOVER TERMINATE command to terminate the DRF address space.

An example of using the /RECOVER TERMINATE command is shown in “Example for /RECOVER TERMINATE Command” on page 560

Examples

Examples for /RECOVER ADD Commands

The COMMAND IN PROGRESS message is issued for /RECOVER ADD commands but is not shown in the following examples.

Example 1 for /RECOVER ADD Command

In the following example, a /RECOVER ADD STALOCAL command is issued for full function database data sets. Following a successful recovery of the database data sets, the database is started on the IMS that runs the recovery.

```
/REC ADD STALOCAL DBDS DBNAME1 DDNAME1 DBNAME2 DDNAME2
DFS4299I FRD6011I THE FOLLOWING ENTRIES ARE ADDED TO THE RECOVERY LIST:
DFS4299I FRD6003I DBNAME1 DDNAME1
DFS4299I FRD6003I DBNAME2 DDNAME2
```

Example 2 for /RECOVER ADD Command

In the following example, a /RECOVER ADD command is issued for full function database data sets. One of the database data sets is not registered in RECON and is rejected.

```
/RECOVER ADD DBDS DBNAME1 DDNAME1 DBNAME1 DDNAME2
DFS4299I FRD6011I THE FOLLOWING ENTRIES ARE ADDED TO THE RECOVERY LIST:
DFS4299I FRD6003I DBNAME1 DDNAME1
DFS4299I FRD6010W UNABLE TO ADD TO RECOVERY LIST, NOT FOUND IN RECON, DBNAME2 DDNAME2
```

Example 3 for /RECOVER ADD Command

In the following example, a /RECOVER ADD command is issued for full function database data set. The database that database data set belongs to is still authorized to two IMSs.

```
/RECOVER ADD DBDS DBNAME1 DDNAME1
DFS4299I FRD6011I THE FOLLOWING ENTRIES ARE ADDED TO THE RECOVERY LIST:
DFS4299I FRD6003I DBNAME1 DDNAME1
DFS4299I FRD6003I DBNAME1 DDNAME1 AUTHORIZED BY IMS1
DFS4299I FRD6003I DBNAME1 DDNAME1 AUTHORIZED BY IMS2
```

Example 4 for /RECOVER ADD Command

Databases can be specified as a whole with the /RECOVER ADD DB command. In the example below, a full function database and a Fast Path database have all their database data sets and areas, respectively, added to the recovery list.

```
/REC ADD DB FFDB1 FPDB2
DFS4299I FRD6011I THE FOLLOWING ENTRIES ARE ADDED TO THE RECOVERY LIST:
DFS4299I FRD6003I FFDB1 DDNAME1
DFS4299I FRD6003I FFDB1 DDNAME2
DFS4299I FRD6003I DBAREA3 DDAREA3
DFS4299I FRD6003I DBAREA4 DDAREA4
```

Example 5 for /RECOVER ADD Command

If more than one DATAGROUP is specified, the group names must be separated by at least one blank space. In the following example, a /RECOVER ADD command is issued for two database groups.

```
/REC ADD DATAGROUP GRPNAME1 GRPNAME2
DFS4299I FRD6011I THE FOLLOWING ENTRIES ARE ADDED TO THE RECOVERY LIST:
DFS4299I FRD6003I DBNAME3 DDNAME3
DFS4299I FRD6003I DBNAME4 DDAREA4
DFS4299I FRD6003I DBNAME5 DDNAME5
DFS4299I FRD6003I DBNAME6 DDAREA6
DFS4299I FRD6003I DBNAME7 DDNAME7
DFS4299I FRD6003I DBNAME8 DDAREA8
```

Example 6 for /RECOVER ADD Command

If one or more RECOVGRP is specified, the group names must be separated by at least one blank. In the following example, a /RECOVER ADD command is issued for two recovery groups.

```
/REC ADD RECOVGRP GRPNAME1 GRPNAME2
DFS4299I FRD6011I THE FOLLOWING ENTRIES ARE ADDED TO THE RECOVERY LIST:
DFS4299I FRD6003I DBNAMEA DDNAMEA
DFS4299I FRD6003I DBNAMEB DDAREAB
DFS4299I FRD6003I DBNAMEC DDNAMEC
DFS4299I FRD6003I DBNAMED DDAREAD
DFS4299I FRD6003I DBNAMEE DDNAMEE
DFS4299I FRD6003I DBNAMEF DDAREAF
```

Examples for /RECOVER REMOVE Command

The command IN PROGRESS message is issued for /RECOVER commands but is not shown in the following examples.

Example 1 for /RECOVER REMOVE Command

As with the /RECOVER ADD command, full function database data sets and Fast Path areas are specified with the /RECOVER REMOVE DBDS command. With the DBDS option, each full function database data set must be specified as an ordered pair. Each element must be separated by at least one blank space. The first element of the pair is the database name. The second element is the DDNAME. In the example below, a /RECOVER REMOVE command is issued for a single full function database data set.

```
/RECOVER REMOVE RCVTOKEN DFS00001 DBDS DBNAME1 DDNAME1

DFS4299I FRD6016I THE FOLLOWING ENTRIES WERE REMOVED FROM THE RECOVERY LIST:
DFS4299I FRD6003I DBNAME1 DDNAME1
```

If more than one full function database data set is specified in a /RECOVER REMOVE DBDS command, each dbname/ddname ordered pair must be separated by at least one blank space.

```
/RECOVER REMOVE RCVTOKEN DFS00001 DBDS DBNAME1 DDNAME1 DBNAME3 DDNAME3

DFS4299I FRD6016I THE FOLLOWING ENTRIES WERE REMOVED FROM THE RECOVERY LIST:
DFS4299I FRD6003I DBNAME1 DDNAME1
DFS4299I FRD6003I DBNAME3 DDNAME3
```

Example 2 for /RECOVER REMOVE Command

In the following example, a /RECOVER REMOVE command is issued for a single Fast Path area that was not added to the recovery list.

```
/REC REMOVE RCVTOKEN DFS00001 AREA DDAREA1

DFS4299I FRD6018W UNABLE TO REMOVE AREA DDAREA1: NOT IN RECOVERY LIST
```

Example 3 for /RECOVER REMOVE Command

In the following example, a /RECOVER REMOVE command is issued for a full function database and Fast Path database. All full function database data sets and Fast Path areas making up the two databases are removed from the recovery list.

```
/REC REMOVE RCVTOKEN DFS00001 DB FFDB1 FFDB2

DFS4299I FRD6016I THE FOLLOWING ENTRIES WERE REMOVED FROM THE RECOVERY LIST:
DFS4299I FRD6003I FFDB1 DDNAME1
DFS4299I FRD6003I FFDB1 DDNAME2
DFS4299I FRD6003I DBAREA3 DDAREA3
DFS4299I FRD6003I DBAREA4 DDAREA4
```

Example 4 for /RECOVER REMOVE Command

In the following example, a /RECOVER REMOVE command is issued to stop recovery for the entire recovery list.

```
/REC REMOVE RCVTOKEN DFS00001 ALLENT

DFS4299I FRD6016I THE FOLLOWING ENTRIES WERE REMOVED FROM THE RECOVERY LIST:
DFS4299I FRD6003I DBNAME1 DDNAME1
DFS4299I FRD6003I DBNAME2 DDNAME2
DFS4299I FRD6003I DBAREA1 DDAREA1
DFS4299I FRD6003I DBNAME3 DDNAME3
DFS4299I FRD6003I DBAREA4 DDAREA4
DFS4299I FRD6003I DBNAME5 DDNAME5
DFS4299I FRD6003I DBAREA6 DDAREA6
DFS4299I FRD6003I DBAREA7 DDAREA7
DFS4299I FRD6003I DBAREA8 DDAREA8
DFS4299I FRD6017I RECOVERY LIST IS NOW EMPTY
```

Examples for /RECOVER START Command

The command IN PROGRESS message is issued for /RECOVER commands but is not shown in the following examples.

Example 1 for /RECOVER START Command

In the following example, /RECOVER START initiates recovery for the database data sets and areas from previous examples. Recovery continues until it completes or until one of the database data sets or areas is operable.

```
/REC START RCVTOKEN RCVTKN1 ERRORCONT
DFS4299I FRD6021I RECOVERY STARTED FOR:
DFS4299I FRD6003I DBNAME1 DDNAME1
DFS4299I FRD6003I DBNAME2 DDNAME2
DFS4299I FRD6003I DBAREA1 DDAREA1
DFS4299I FRD6003I DBNAME3 DDNAME3
DFS4299I FRD6003I DBAREA4 DDAREA4
DFS4299I FRD6003I DBAREA5 DDAREA5
DFS4299I FRD6003I DBAREA6 DDAREA6
DFS4299I FRD6003I DBNAME7 DDNAME7
DFS4299I FRD6003I DBAREA8 DDAREA8
```

Example 2 for /RECOVER START Command

In the following example, /RECOVER START RCVTOKEN initiates recovery for the database data sets and areas owned by the recovery token RCVTKN2.

```
/REC START RCVTOKEN RCVTKN2
DFS4299I FRD6021I RECOVERY STARTED FOR:
DFS4299I FRD6003I DBNAME1 DDNAME1
DFS4299I FRD6003I DBNAME2 DDNAME2
DFS4299I FRD6003I DBAREA1 DDAREA1
DFS4299I FRD6003I DBNAME3 DDNAME3
DFS4299I FRD6003I DBAREA4 DDAREA4
DFS4299I FRD6003I DBAREA5 DDAREA5
DFS4299I FRD6003I DBAREA6 DDAREA6
DFS4299I FRD6003I DBNAME7 DDNAME7
DFS4299I FRD6003I DBAREA8 DDAREA8
```

Example 3 for /RECOVER START Command

In the example below, /RECOVER START RCVTOKEN OFFLINE READNUM 6 initiates recovery for the database data sets and areas from previous examples. Recovery will not continue if any error is detected for any member of the recovery list. The database data sets and areas remain offline after recovery completes.

```
/RECOVER START RCVTOKEN RCVTKN2 OFFLINE READNUM 6
DFS4299I FRD6021I RECOVERY STARTED FOR:
DFS4299I FRD6003I DBNAME1 DDNAME1
DFS4299I FRD6003I DBNAME2 DDNAME2
DFS4299I FRD6003I DBAREA1 DDAREA1
DFS4299I FRD6003I DBNAME3 DDNAME3
DFS4299I FRD6003I DBAREA4 DDAREA4
DFS4299I FRD6003I DBAREA5 DDAREA5
DFS4299I FRD6003I DBAREA6 DDAREA6
DFS4299I FRD6003I DBNAME7 DDNAME7
DFS4299I FRD6003I DBAREA8 DDAREA8
```

Example 4 for /RECOVER START Command

In the following example, /RECOVER START RCVTOKEN ERRORCONT RCVTIME *time-stamp* is issued. TSR continues until it completes or until one of the database data sets or areas undergoing recovery encounters an error. After recovery completes, a message is issued listing each database data set and area successfully recovered.

```
3 /REC START RCVTOKEN RCVTKN2 ERRORABORT RCVTIME '020011015257' NOCHECK
3 DFS4299I FRD6021I RECOVERY STARTED FOR RCVTKN2, ERRORABORT, TSR
3 DFS4299I FRD6003I DBNAME1 DDNAME1
3 DFS4299I FRD6003I DBNAME2 DDNAME2
3 DFS4299I FRD6003I DBAREA1 DDAREA1
3 DFS4299I FRD6003I DBNAME3 DDNAME3
3 DFS4299I FRD6003I DBAREA4 DDAREA4
3 DFS4299I FRD6003I DBAREA5 DDAREA5
3 DFS4299I FRD6003I DBAREA6 DDAREA6
3 DFS4299I FRD6003I DBNAME7 DDNAME7
3 DFS4299I FRD6003I DBAREA8 DDAREA8
3     ...
3 DFS4299I FRD4031I DATASET RESTORE COMPLETE:  DBNAME1 DDNAME1
3 DFS4299I FRD4031I DATASET RESTORE COMPLETE:  DBNAME2 DDNAME2
3 DFS4299I FRD4031I DATASET RESTORE COMPLETE:  DBAREA1 DDAREA1
3 DFS4299I FRD4031I DATASET RESTORE COMPLETE:  DBNAME3 DDNAME3
3 DFS4299I FRD4031I DATASET RESTORE COMPLETE:  DBAREA4 DDAREA4
3 DFS4299I FRD4031I DATASET RESTORE COMPLETE:  DBAREA5 DDAREA5
3 DFS4299I FRD4031I DATASET RESTORE COMPLETE:  DBAREA6 DDAREA6
3 DFS4299I FRD4031I DATASET RESTORE COMPLETE:  DBNAME7 DDNAME7
3 DFS4299I FRD4031I DATASET RESTORE COMPLETE:  DBAREA8 DDAREA8
3 DFS4277I RECOVERY COMPLETE FOR:  RCVTKN2
```

Example 5 for /RECOVER START Command

In the following example, /RECOVER START RCVTOKEN RCVTIME*time-stamp* PITR is issued. Point-in-time recovery will continue until it completes or until one of the database data sets encounters an error. A message is issued listing the database data sets and areas that were not in the recovery list, but might need recovery using point-in-time recovery.

```
3 /REC START RCVTOKEN RCVTKN2 RCVTIME '020011015257' PITR
3 DFS4299I FRD6021I RECOVERY STARTED FOR RCVTKN2, ERRORCONT, PITR
3 DFS4299I FRD6003I DBNAME1 DDNAME1
3 DFS4299I FRD6003I DBNAME2 DDNAME2
3 DFS4299I FRD6003I DBAREA1 DDAREA1
3 DFS4299I FRD6003I DBNAME3 DDNAME3
3 DFS4299I FRD6003I DBAREA4 DDAREA4
3 DFS4299I FRD6003I DBAREA5 DDAREA5
3 DFS4299I FRD6003I DBAREA6 DDAREA6
3 DFS4299I FRD6003I DBNAME7 DDNAME7
3 DFS4299I FRD6003I DBAREA8 DDAREA8
3     ...
3 DFS4299I FRD6024A GROUP MEMBER DBNAME9 DDNAME9 NOT IN RECOVERY LIST: MEMBER OF A GROUP
```

```
DFS4299I FRD6024A GROUP MEMBER DBNAMEA DDNAMEA NOT IN RECOVERY LIST: MEMBER OF A GROUP
DFS4299I FRD6024A GROUP MEMBER DBAREAB DDAREAB NOT IN RECOVERY LIST: MEMBER OF A GROUP
DFS4299I FRD6024A GROUP MEMBER DBNAMEC DDNAMEC NOT IN RECOVERY LIST: MEMBER OF A GROUP
    ...
DFS4299I FRD4031I DATASET RESTORE COMPLETE:  DBNAME1 DDNAME1
DFS4299I FRD4031I DATASET RESTORE COMPLETE:  DBNAME2 DDNAME2
DFS4299I FRD4031I DATASET RESTORE COMPLETE:  DBAREA1 DDAREA1
DFS4299I FRD4031I DATASET RESTORE COMPLETE:  DBNAME3 DDNAME3
DFS4299I FRD4031I DATASET RESTORE COMPLETE:  DBAREA4 DDAREA4
DFS4299I FRD4031I DATASET RESTORE COMPLETE:  DBAREA5 DDAREA5
DFS4299I FRD4031I DATASET RESTORE COMPLETE:  DBAREA6 DDAREA6
DFS4299I FRD4031I DATASET RESTORE COMPLETE:  DBNAME7 DDNAME7
DFS4299I FRD4031I DATASET RESTORE COMPLETE:  DBAREA8 DDAREA8
DFS4277I RECOVERY COMPLETE FOR:  RCVTKN2
```

Examples for /RECOVER STOP Command

The COMMAND IN PROGRESS message is issued for /RECOVER commands but is not shown in the following examples.

Example 1 for /RECOVER STOP

In the following example, a /RECOVER STOP command is issued to stop recovery for the entire recovery list.

```
/REC STOP ALLENT
DFS4299I FRD6032I THE FOLLOWING ENTRIES WILL HAVE RECOVERY STOPPED:
DFS4299I FRD6003I DBNAME1 DDNAME1
DFS4299I FRD6003I DBNAME2 DDNAME2
DFS4299I FRD6003I DBAREA1 DDAREA1
DFS4299I FRD6003I DBNAME3 DDNAME3
DFS4299I FRD6003I DBAREA4 DDAREA4
DFS4299I FRD6003I DBNAME5 DDNAME5
DFS4299I FRD6003I DBAREA6 DDAREA6
DFS4299I FRD6003I DBAREA7 DDAREA7
DFS4299I FRD6003I DBAREA8 DDAREA8
DFS4299I FRD6033I ALL ENTRIES IN RECOVERY LIST, ARE BEING STOPPED
```

Example 2 for /RECOVER STOP

In this example, a /RECOVER STOP ALLENT SAVE is issued after a /RECOVER START command.

```
/REC STOP ALLENT SAVE
DFS4299I FRD6032I THE FOLLOWING ENTRIES WILL HAVE RECOVERY STOPPED:
DFS4299I FRD6003I DBNAME1 DDNAME1
```

Example 3 for /RECOVER STOP

In this example, a /RECOVER STOP ALLENT command is issued with no recovery in progress.

```
/REC STOP ALLENT
DFS4299I FRD6031E UNABLE TO STOP ALLENT: RECOVERY NOT IN PROGRESS
```

Example for /RECOVER TERMINATE Command

In this example, a /RECOVER TERMINATE command is issued with no recovery in progress.

```
/RECOVER TERMINATE
DFS4299I FRD4202I DATABASE RECOVERY DATA MANAGER TERMINATION COMPLETE
```

Chapter 48. /RELEASE

Format

Environments and Keywords

Table 134 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keyword can be issued.

Table 134. Valid Environments for the /RELEASE Command and Keyword

Command / Keyword	DB/DC	DBCTL	DCCTL
/RELEASE	X		X
CONVERSATION	X		X

Usage

/RELEASE resumes a conversation that was previously saved by means of the /HOLD command.

The last message sent to the terminal before /HOLD was entered is sent to the terminal again.

/RELEASE is not valid from an LU 6.2 device. LU 6.2 communications cannot release a conversation, whether started by itself or by another communications protocol.

If global resource information is kept in Resource Manager, /RELEASE updates the conversation globally in Resource Manager. If global resource information is not kept in Resource Manager, /RELEASE updates the conversation locally.

CONVERSATION
Specifies the 4-digit identification (including leading zeros) of the conversation to be resumed; CONVERSATION conv# is the 4-digit identification conv# that was provided when the conversation was previously held.

Example for /RELEASE Command

Entry ET:

```
/RELEASE CONVERSATION 0001
```

Response ET:

IMS does not respond to this command except to resend the last message.

Explanation: Conversation 0001 has been released and can be resumed by the terminal operator.

© Copyright IBM Corp. 1974, 2003

Chapter 49. /RESET

Format

Environments

Table 135 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command can be issued.

Table 135. Valid Environments for the /RESET Command

Command	DB/DC	DBCTL	DCCTL
/RESET	X		X

Usage

/RESET eliminates the preset mode established by the /SET command.

Example for /RESET Command

Entry ET:

```
/RESET
```

Response ET:

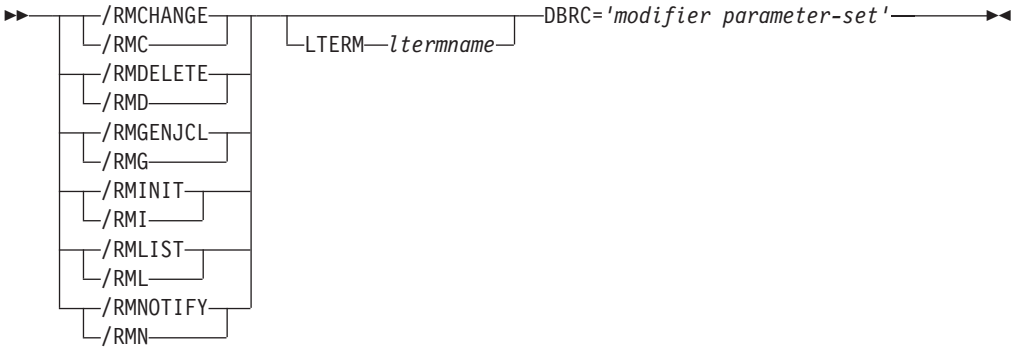
```
DFS058I  RESET COMMAND COMPLETED
```

Explanation: The preset mode is no longer in effect.

© Copyright IBM Corp. 1974, 2003

Chapter 50. /RMxxxxxx

Format



Environments and Keywords

Table 136 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keyword can be issued.

Table 136. Valid Environments for the /RMxxxxxx Command and Keyword

Command / Keyword	DB/DC	DBCTL	DCCTL
/RMxxxxxx	X	X	X
LTERM	X		X

Usage

The /RMxxxxxx commands are multisegment commands that call functions of IMS Database Recovery Control (DBRC). These commands allow the IMS master terminal operator or an authorized terminal operator to run certain DBRC utility functions online. Output is limited to what can be put in a 4 KB buffer (8 KB for /RMLIST).

All /RMxxxxxx formats require an EOM indication to denote end-of-message. An EOS indication must be included for all segments, if any, that precede the last segment. See “Multisegment Command Input” on page 8 for more detail on using EOS and EOM. If comments are included with the /RMxxxxxx commands, they must be enclosed in asterisks.

If a failure other than the loss of both RECON data sets occurs while DBRC is processing an online command, DBRC makes the command unavailable for the remaining time the IMS online region is running. (After determining and correcting the cause of the original failure, the command can be made available again by resubmitting the online command with the RESET parameter specified in the parameter set.) It is the **verb**, rather than the modifier, level of the command that DBRC makes unavailable. That is, if a DBRC INIT.DB command fails, DBRC makes *all* INIT commands unavailable. DBRC sends an error message to the originating terminal when the command fails. You can still issue the failing command from other IMS online regions.

© Copyright IBM Corp. 1974, 2003

Exception: DBRC does not make GENJCL commands unavailable because the GENJCL.ARCHIVE command is needed for automatic archiving.

DBRC does not remember command failures across IMS restarts because it assumes that you will correct the error before restarting IMS.

Table 137 lists the DBRC commands that are supported online and describes the utility function of each command.

Table 137. Functions of the DBRC Commands Supported Online

Command	Utility Function
/RMCHANGE	Changes or modifies information in the RECON data set
/RMDELETE	Deletes information in the RECON data set
/RMGENJCL	Generates JCL for: • IMS Change Accumulation utility • IMS Log Archive utility • IMS Log Recovery utility • IMS Database Image Copy utility • Database Image Copy 2 • IMS Online Database Image Copy utility • Database Recovery utility • User-defined output
/RMINIT	Creates records in the DBRC RECON data set
/RMLIST	Lists information contained in the RECON data set
/RMNOTIFY	Adds information to the RECON data set

LTERM

Specifies the logical terminal designated for output. If you omit the LTERM keyword, the output destination is the input terminal.

Recommendation: Because some of the DBRC commands generate a large amount of output, especially the /RMGENJCL and /RMLIST commands, direct the output to a printer.

modifier

The DBRC modifier for the function specified.

Table 138 lists the DBRC modifiers and the /RMxxxxxx commands with which the modifiers can be issued.

Table 138. DBRC Modifiers for the /RMxxxxxx Commands

Modifier	CHANGE	DELETE	GENJCL	INIT	LIST	NOTIFY
ADS	X	X		X		
ALLOC		X				X
ARCHIVE			X			
BKOUT	X	X			X	X
CA	X	X	X	X		X
CAGRP	X	X		X	X	
CLOSE			X			
DB	X	X		X	X	
DBDS	X	X		X	X	

Table 138. DBRC Modifiers for the /RMxxxxxx Commands (continued)

Modifier	CHANGE	DELETE	GENJCL	INIT	LIST	NOTIFY
DBDSGRP	X	X		X	X	
GSG		X		X	X	
HISTORY					X	
IC	X	X	X	X		X
LOG		X			X	
OIC			X			
PART				X		
PRILOG	X					X
RECON	X			X	X	
RECOV		X	X			X
REORG		X				X
SECLOG	X					X
SG	X	X		X		
SUBSYS	X	X			X	X
UIC	X	X				X
USER			X			

parameter-set

Represents the required and optional parameters that will be passed to DBRC. For a full description of the DBRC commands, modifiers, and parameters, see *IMS Version 9: DBRC Guide and Reference*.

Examples

Example for /RMCHANGE Command

Entry ET (with comments):

```
/RMCHANGE DBRC='DB DBD(DIVNTZ04) SHARELVL(3)'. *COMMENT*.
```

Response ET:

```
CHANGE.DB DBD(DIVNTZ04) SHARELVL(3)
DSP0203I  COMMAND COMPLETED WITH CONDITION CODE 00
DSP0220I  COMMAND COMPLETION TIME
DSP0211I  COMMAND PROCESSING COMPLETE
DSP0211I  HIGHEST CONDITION CODE = 00
DSP0058I  RMC COMMAND COMPLETED
```

Entry ET:

```
/RMCHANGE DBRC='DBDS DBD(DIVNTZ04) DDN(DBHVSAM1) ICON'.
```

Response ET:

```
CHANGE.DBDS DBD(DIVNTZ04) DDN(DIVNTZ04) ICON
DSP0203I  COMMAND COMPLETED WITH CONDITION CODE 00
DSP0220I  COMMAND COMPLETION TIME
DSP0211I  COMMAND PROCESSING COMPLETE
DSP0211I  HIGHEST CONDITION CODE = 00
DSP0058I  RMC COMMAND COMPLETED
```

Entry ET:

```
/RMC DBRC='DBDS DBD(DEDBJN21) AREA(DB21AR7) VSO PREOPEN'.
```

Response ET:

```
DFS000I CHANGE.DBDS DBD(DEDBJN21) AREA(DB21AR7) VSO PREOPEN
DFS000I DSP0203I  COMMAND COMPLETED WITH CONDITION CODE 00
```

Example for /RMDELETE Command

Entry ET (with comments):

```
/RMDELETE DBRC='DB DBD(DIVNTZ04)'. *COMMENT*.
```

Response ET:

```
DELETE.DB DBD(DIVNTZ04)
DSP0203I COMMAND COMPLETED WITH CONDITION CODE 00
DSP0220I  COMMAND COMPLETION TIME
DSP0211I  COMMAND PROCESSING COMPLETE
DSP0211I  HIGHEST CONDITION CODE = 00
DSP0058I  RMD COMMAND COMPLETED
```

Example for /RMGENJCL Command

Entry ET (with comments):

```
/RMGENJCL LTERM SMITH DBRC='IC DBD(HDAMVSAM) DDN(DD1) LIST'. *END OF DAY MESSAGE*.
```

Response ET:

```
DSP058I RMG COMMAND COMPLETED
```

Response LTERM SMITH:

```
GENJCL.IC DBD(HDAMVSAM) DDN(DD1)
//IC135607   JOB
//IC        EXEC PGM=DFSRRC00,PARM='ULU,DFSUDMP0',REGION=800K
//*
//*     THIS JCL ORIGINATES FROM THE USER'S 'JCLPDS' LIBRARY.
//*     %KEYWORDS ARE REPLACED BY THE GENJCL FUNCTION OF
//*     THE IMS DATABASE RECOVERY CONTROL FEATURE.
//*
//*                 JCL FOR IMAGE COPY.
//*
//SYSPRINT  DD SYSOUT=A
//RECON1    DD DSN=POCONO1,DISP=SHR
//RECON2    DD DSN=POCONO2,DISP=SHR
//IMS        DD DSN=IMS.DBDLIB,DISP=SHR
//DD1    DD DSN=HDAMVASM,DCB=BUFNO=10,DISP=OLD
//DATAOUT1  DD DSN=IMS.HDAMVSAM.DD1.IC.ICDD1,UNIT=3400,
//             VOL=(PRIVATE,,,1,SER=(******)),
//             LABEL=(1,SL),
//             DISP=(NEW,KEEP),DCB=BUFNO=10
//DFSVSAMP  DD *
1024,2
4096,4
//SYSIN     DD *
D1 HDAMVSAM DD1      DATAOUT1
DSP0203I  COMMAND COMPLETED WITH CONDITION CODE 00
DSP0220I  COMMAND COMPLETION TIME
DSP0211I  COMMAND PROCESSING COMPLETE
DSP0211I  HIGHEST CONDITION CODE = 00
DSP0058I  RMG COMMAND COMPLETED
```

Example for /RMINIT Command

Entry ET (with comments):

```
/RMINIT DBRC='DB DBD(DIVNTZ04) SHARELVL(3)'. *COMMENT*.
```

Response ET:

```
INIT.DB DBD(DIVNTZ04) SHARELVL(3)
DSP0203I  COMMAND COMPLETED WITH CONDITION CODE 00
DSP0220I  COMMAND COMPLETION TIME
DSP0211I  COMMAND PROCESSING COMPLETE
DSP0211I  HIGHEST CONDITION CODE = 00
DSP0058I  RMI COMMAND COMPLETED
```

Example for /RMLIST Command

Entry ET (with comments):

```
/RMLIST DBRC='DB DBD(DIVNTZ04)'. *LAST COMMENT*.
```

Response ET:

```
LIST.DB DBD(DIVNTZ04)

           RECON1 DSN=RECONH1

--------------------------------------------------------------------------------
 DB
  DBD=DINNTZ04                IRLM=*NULL     DMB#=769          TYPE=IMS
  SHARE LEVEL=0

    FLAGS:                                 COUNTERS:
             BACKOUT NEEDED=OFF              RECOVERY NEEDED COUNT=0
                  READ ONLY=OFF             IMAGE COPY NEEDED COUNT=1
     PROHIBIT AUTHORIZATION=OFF              AUTHORIZED SUBSYSTEMS=0
                RECOVERABLE=YES            HELD AUTHORIZATION STATE=0
                                                        EEQE COUNT=0

DSP0180I  NUMBER OF RECORDS LISTED IS 1
DSP0203I  COMMAND COMPLETED WITH CONDITION CODE 00
DSP0220I  COMMAND COMPLETION TIME
DSP0211I  COMMAND PROCESSING COMPLETE.
DSP0211I  HIGHEST CONDITION CODE = 00
DSP0058I  RML COMMAND COMPLETED
```

Entry ET:

```
/RMLIST DBRC='DB DBD(DIVNTZ04) DBDS'
```

Response ET:

```
LIST.DB DBD(DIVNTZ04) DBDS
           RECON1 DSN=RECONH1
--------------------------------------------------------------------------------
 DB
  DBD=DIVNTZ04               IRLMID=*NULL     DMB#=769    TYPE=IMS
  SHARE LEVEL=0
    FLAGS:                                   COUNTERS:
             BACKOUT NEEDED=OFF               RECOVERY NEEDED COUNT=0
                  READ ONLY=OFF              IMAGE COPY NEEDED COUNT=1
     PROHIBIT AUTHORIZATION=OFF               AUTHORIZED SUBSYSTEMS=0
                RECOVERABLE=YES             HELD AUTHORIZATION STATE=0
                                                          EEQE COUNT=0

--------------------------------------------------------------------------------
 DBDS   DSN     =JDSGIIZ4                                        DBDS SEQ=1   IMS
        DBD=DIVNTZ04 DDN=DBHVSAM1 DSID=01 DBORG=HISAM  DSORG=VSAM DSLOG SEQ=00
        CA GRPNAME=**NULL** GENMAX=3    IC AVAIL=0    IC USED=0
                IC NEEDED-ON                    RECOV-OFF
        NOREUSE ICJCL=ICJCL    OICJCL=OICJCL   RECOVJCL=RECOVJCL
--------------------------------------------------------------------------------
 DBDS   DSN     =JDSGIIZ4                                        DBDS SEQ=2   IMS
        DBD=DIVNTZ04 DDN=DBHVSAM2 DSID=01 DBORG=HISAM  DSORG=VSAM DSLOG SEQ=00
        CA GRPNAME=**NULL** GENMAX=3    IC AVAIL=0    IC USED=0
```

```
               IC NEEDED-OFF                  RECOV-OFF
        NOREUSE ICJCL=ICJCL    OICJCL=OICJCL   RECOVJCL=RECOVJCL
DSP0180I  NUMBER OF RECORDS LISTED IS 3
DSP0203I  COMMAND COMPLETED WITH CONDITION CODE 00
DSP0220I  COMMAND COMPLETION TIME
DSP0211I  COMMAND PROCESSING COMPLETE.
DSP0211I  HIGHEST CONDITION CODE = 00
DSP0058I  RML COMMAND COMPLETED
```

Entry ET (With Comments):

```
/RML DBRC='DBDS DBD(DEDBJN21) AREA(DB21AR0)'. *VSO AREA*.
```

Response ET:

```
LIST.DBDS DBD(DEDBJN21) AREA(DB21AR0)
------------------------------------------------------------------------
 DBDS
  DBD=DEDBJN21  AREA=DB21AR0                                    TYPE=FP
  SHARE LEVEL=1             DSID=001 DBORG=DEDB   DSORG=VSAM
  GSG NAME=**NULL**           USID=0000000002
  AUTHORIZED USID=0000000002  RECEIVE USID=0000000002  MAX USID=0000000002
  RECEIVE NEEDED USID=0000000000
  CAGRP=**NULL**  GENMAX=5     IC AVAIL=0     IC USED=0     DSSN=0000000
  HSSP IC IN PROCESS=NO          AVAIL         USED         PARTIAL
                                HSSP IC=0     HSSP IC=0     HSSP IC=0
  REUSE           RECOVPD=0      VSO    PREOPEN    PRELOAD
  DEFLTJCL=**NULL**  ICJCL=ICJCL    RECVJCL=ICRCVJCL  RECOVJCL=RECOVJCL
  FLAGS:                            COUNTERS:
    PROHIBIT AUTHORIZATION=OFF        AUTHORIZED SUBSYSTEMS   =1
                                      HELD AUTHORIZATION STATE=7
    IC NEEDED             =OFF        ADS AVAIL #             =1
    RECOV NEEDED          =OFF        REGISTERED ADS #        =1
                                      EEQE COUNT              =0
    TRACKING IN PROGRESS  =NO         RECEIVE NEEDED          =OFF
    OFR REQUIRED          =NO
    TRACKING SUSPENDED    =NO
  ADS LIST:
                                                                CREATE
     -ADS DDN--ADS DSN-                                      -STAT- -RUNNING-
     DB21AR0  DB21AR0                                        AVAIL     NO
   ASSOCIATED SUBSYSTEM INFORMATION:
                              ENCODED
      -SSID-   -ACCESS INTENT-  -STATE-  -SS ROLE-
     SYS3         EXCLUSIVE        7      ACTIVE
  ALLOC
   ALLOC   = 93.076   13:30:35.0*     START   = 93.076   13:30:23.3
   LRID=0000000000000000        DSSN=0000000001     USID=0000000002
  REORG
   RUN     = 93.076   13:23:38.0*     USID=0000000000
 DSP0180I  NUMBER OF RECORDS LISTED IS        3
 DSP0203I  COMMAND COMPLETED WITH CONDITION CODE 00
 DSP0220I  COMMAND COMPLETION TIME 93.076   13:37:36.7
 DSP0211I  COMMAND PROCESSING COMPLETE
 DSP0211I  HIGHEST CONDITION CODE = 00
 DSP0058I  RML COMMAND COMPLETED
```

Entry ET (With Comments):

```
/RML DBRC='DBDS DBD(DEDBJN21) AREA(DB21AR6)'. *DEDB AREA*.
```

Response ET:

```
LIST.DBDS DBD(DEDBJN21) AREA(DB21AR6)
------------------------------------------------------------------------
 DBDS
  DBD=DEDBJN21  AREA=DB21AR6                                    TYPE=FP
  SHARE LEVEL=1             DSID=007 DBORG=DEDB   DSORG=VSAM
```

```
 GSG NAME=**NULL**            USID=0000000002
 AUTHORIZED USID=0000000002  RECEIVE USID=0000000002  MAX USID=00000000
 RECEIVE NEEDED USID=0000000000
 CAGRP=**NULL**  GENMAX=5     IC AVAIL=0     IC USED=0     DSSN=0000000
 HSSP IC IN PROCESS=NO          AVAIL         USED         PARTIAL
                               HSSP IC=1     HSSP IC=0     HSSP IC=0
 REUSE           RECOVPD=0      NOVSO  NOPREOPEN  NOPRELOAD
 DEFLTJCL=**NULL**  ICJCL=ICJCL    RECVJCL=ICRCVJCL  RECOVJCL=RECOVJCL
 FLAGS:                            COUNTERS:
   PROHIBIT AUTHORIZATION=OFF        AUTHORIZED SUBSYSTEMS   =0
                                     HELD AUTHORIZATION STATE=0
   IC NEEDED             =OFF        ADS AVAIL #             =0
   RECOV NEEDED          =ON         REGISTERED ADS #        =1
                                     EEQE COUNT              =0
   TRACKING IN PROGRESS  =NO         RECEIVE NEEDED          =OFF
   OFR REQUIRED          =NO
   TRACKING SUSPENDED    =NO
 ADS LIST:
                                                                 CREATE
   -ADS DDN--ADS DSN-                                       -STAT- -RUNNING
   DB21AR6  DB21AR6                                         UNAVAIL   NO
ALLOC
 ALLOC   = 93.076   13:32:05.0*     START   = 93.076   13:30:23.3
 LRID=0000000000000000        DSSN=0000000001     USID=0000000002
 DEALLOC = 93.076   13:34:21.2      DEALLOC LRID=0000000000000000
REORG
 RUN     = 93.076   13:24:12.5*     USID=0000000000
                          AVAILABLE DATA SET
                       DBD=DEDBJN21 DDN=DB21AR6
 IMAGE
* CREATE  = 93.076   13:22:52.8*     HSSP
 IC1
  DSN=IC21AR6
DSP0180I  NUMBER OF RECORDS LISTED IS        4
DSP0203I  COMMAND COMPLETED WITH CONDITION CODE 00
DSP0220I  COMMAND COMPLETION TIME 93.076   13:38:21.0
DSP0211I  COMMAND PROCESSING COMPLETE
DSP0211I  HIGHEST CONDITION CODE = 00
DSP0058I  RML COMMAND COMPLETED
```

Example for /RMNOTIFY Command

Entry ET (with comments):

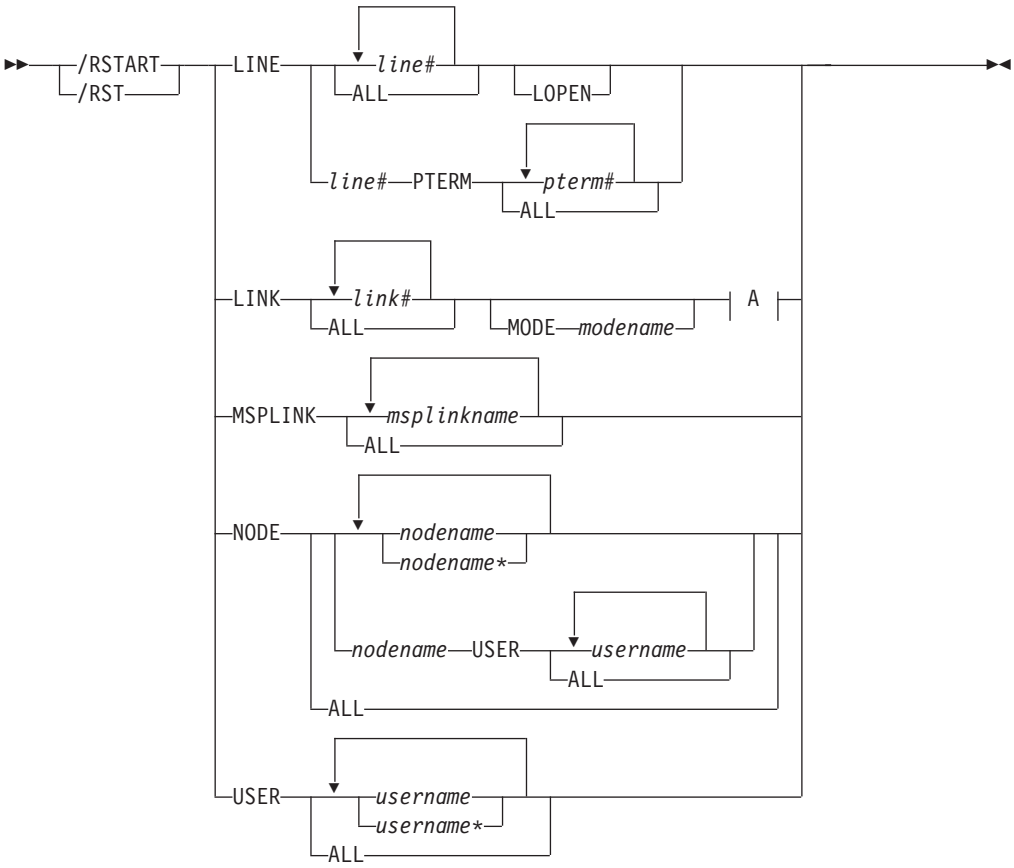
```
/RMNOTIFY DBRC='SUBSYS SSID(IMSB) IRLMID(IRLM1) NORMAL'. *END OF DAY MESSAGE*.
```

Response ET:

```
NOTIFY.SUBSYS SSID(IMSB) IRLMID(IRLM1) NORMAL
DSP0203I  COMMAND COMPLETED WITH CONDITION CODE 00
DSP0220I  COMMAND COMPLETION TIME
DSP0211I  COMMAND PROCESSING COMPLETE.
DSP0211I  HIGHEST CONDITION CODE = 00
DSP0058I  RMN COMMAND COMPLETED
```

Chapter 51. /RSTART

Format



A:

Environments and Keywords

Table 139 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 139. Valid Environments for the /RSTART Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/RSTART	X		X
CONTINUOUS	X		X
LINE	X		X
LINK	X		X
LOPEN	X		X

© Copyright IBM Corp. 1974, 2003

Table 139. Valid Environments for the /RSTART Command and Keywords (continued)

Command / Keywords	DB/DC	DBCTL	DCCTL
MODE	X		X
MSPLINK	X		X
NODE	X		X
PTERM	X		X
USER	X		X

Usage

/RSTART starts lines, lines and physical terminals, logical links, nodes, and users when you do not want to reset all associated conditions such as a conversation or special operating mode.

/RSTART checks the validity of all parameters entered by the terminal operator. If an error is detected on parameters that are independent of one another, only the invalid parameters are flagged as being in error and the /RSTART command processes the rest of the parameters.

/RSTART can be used to reset conditions previously established by the /START, /STOP, /PSTOP, /PURGE, /MONITOR, /COMPT, or /RCOMPT command.

LINE
specifies the line or line/physical terminal to be started with all terminals on the line in the same mode of operation they were in when they were stopped. /RSTART LINE allows all input, output, and queuing to start on the line and take the line out of response mode if the line was in this mode. /RSTART LINE PTERM does not reset line response mode, but does reset the BTAM-attached 3270 terminal response mode and looptest mode.

LOPEN
LOPEN enables stopped and idle remote BTAM lines. Enter the /RSTART LINE LOPEN command before any /RSTART LINE PTERM command to avoid having a line that is stopped and idle reset before it can be enabled again. If the line is not stopped or process stopped, and idle, or if enabling is not applicable, the LOPEN keyword is ignored and processing continues.

/RSTART LINE and /RSTART NODE cannot reset terminal response mode if Fast Path is active for a specified physical terminal or node. /DEQUEUE must be entered to discard Fast Path output before using /RSTART.

LINK
Specifies the logical links to be started, either individually or all at once. Communication between IMS systems does not begin until a /START LINK command is entered in both systems for BTAM, CTC, or MTM link, or in either one of the systems for a VTAM link. /RSTART LINK will be rejected unless the link is in process stopped and idled status and the assigned physical link is open, as shown in the /DISPLAY command. /RSTART LINK also re-enables BTAM-attached BSC logical links.

CONTINUOUS
Keeps the link running by sending dummy data blocks when there is no data to be sent. These blocks are discarded when received. This mode of operation eliminates the need for either side to bid for the line, which can

improve the traffic handling capabilities of a high usage link. The block size parameter is the size of the dummy data blocks to be sent.

If the size is not specified, it defaults to 2 bytes. The maximum size of the dummy data blocks to be sent is equal to the size of the link buffer minus 80 bytes.

The CONTINUOUS keyword is ignored if the link is not BTAM. If multiple links are restarted and the CONTINUOUS keyword is specified, only the BISYNC links are restarted in continuous mode.

A /PSTOP LINK command resets continuous mode. If a link is stopped because of an error while in continuous mode, a /RSTART command that does not specify the CONTINUOUS parameter will reset continuous mode.

MODE

The MODE keyword allows you to specify mode table entries to be used when activating an IMS VTAM MSC session. Use of the MODE keyword with non-VTAM links is invalid. If non-VTAM links are referred to specifically in a /RSTART LINK command with the MODE keyword, they will be marked in error.

MSPLINK

Specifies that only MSC VTAM links be reset to allow logons.

NODE

Specifies the VTAM node for which input, output, and queuing will start. The /RSTART NODE nodename USER username command restarts the ISC half-session allocated to *username* for *nodename*. The USER keyword is valid only with the NODE keyword and restarts the specified half-session. If the USER keyword is omitted, all half-sessions of the specified node are affected.

These conditions apply to ISC, dynamic 3600 and dynamic LUP.

The NODE parameter can be generic if the USER keyword is not specified. The generic parameter specifies nodes that already exist. If the node was created temporarily to retain status data and the status conditions have been reset, then the node is deleted at the next simple checkpoint.

If global resource information is not kept in Resource Manager, the /RSTART NODE command allows a node to logon to the local IMS, without resetting local status. If global resource information is kept in Resource Manager, the /RSTART NODE command allows a node to logon to any IMS in the IMSplex, without resetting global node status kept in Resource Manager. If the node no longer has significant status, it is deleted by Resource Manager.

USER

Specifies the USER for which input, output, and queueing are to start. This command starts the USER without resetting conditions such as conversation mode, exclusive mode, and test mode. The /RSTART USER command applies only to dynamic users.

The USER parameter can be generic where the generic parameter specifies already existing users.

If the user structure is temporary and was created solely to retain status that is now reset, the temporary user is deleted at the next simple checkpoint.

If global resource information is not kept in Resource Manager, the /RSTART USER command allows a user to signon to the local IMS. If global resource information is kept in Resource Manager, the /RSTART USER command allows a

user to signon to any IMS in the IMSplex. If the user no longer has significant status in Resource Manager, it is deleted.

Examples

Example 1 for /RSTART Command

Entry ET:

```
/RSTART LINE 4,5,6,7,8,9,10,11
```

Response ET:

```
DFS058I  RSTART COMMAND COMPLETED
```

Response RT:

```
DFS059I  TERMINAL RSTARTED
```

Explanation: LINES 4,5,6,7,8,9,10, and 11 are started.

Example 2 for /RSTART Command

Entry ET:

```
/RSTART LINE 4 5 6 700
```

Response ET:

```
DFS058I  RSTART COMMAND COMPLETED EXCEPT LINE 700
```

Example 3 for /RSTART Command

Entry ET:

```
/RSTART LINE 4 PTERM 1, 2
```

Response ET:

```
DFS058I  RSTART COMMAND COMPLETED
```

Response RT:

```
DFS059I  TERMINAL RSTARTED
```

Explanation: LINE 4 PTERM 1 and 2 are started.

Example 4 for /RSTART Command

Entry ET:

```
/RSTART LINE 4 LOPEN
```

Response RT:

```
DFS058I  RSTART COMMAND COMPLETED
```

Response ET:

```
DFS059I  TERMINAL RSTARTED
```

Explanation: If line 4 is a 3270 remote BTAM line that is idle and stopped, the line is restarted and re-enabled.

Example 5 for /RSTART Command

Entry ET:

```
/RSTART LINK ALL
```

Response ET:

```
DFS058I  RSTART COMMAND COMPLETED
```

Explanation: All of the logical links are started. Communication across the link will not begin until the partner in the remote system is started with the /RSTART LINK command.

Response ET:

```
DFS2168I  CONNECTION ESTABLISHED ON LINK 2
```

Explanation: The connection for communication between two IMS systems is established. The partner link is started with a /RSTART LINK command. After each connection, the message DFS2168 is returned.

Example 6 for /RSTART Command

Entry ET:

```
/RSTART LINK 2
```

Response ET:

```
DFS058I  RSTART COMMAND COMPLETED
```

Explanation: Logical link 2 is started.

Response ET:

```
DFS2168I  CONNECTION ESTABLISHED ON LINK 2
```

Explanation: The two IMS systems are connected.

Example 7 for /RSTART Command

Entry ET:

```
/RSTART LINK 4
```

Response ET:

```
DFS058I  RSTART COMMAND COMPLETED
```

Response ET at Partner System:

```
DFS2160I LINK 4 STARTED BY PARTNER AB NODE WEST
```

Explanation: Link 4 was started at the request of the primary system.

Example 8 for /RSTART Command

Entry ET:

```
/RSTART NODE EAST
```

Response ET:

```
DFS058I  RSTART COMMAND COMPLETED
```

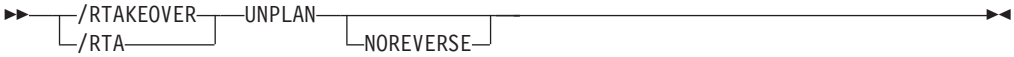
Explanation: The node named EAST is started.

Chapter 52. /RTAKEOVER

Format

/RTAKEOVER for an Active Subsystem

/RTAKEOVER for a Tracking Subsystem



Environments and Keywords

Table 140 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 140. Valid Environments for the /RTAKEOVER Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/RTAKEOVER	X	X	X
DUMPQ	X	X	X
FREEZE	X	X	X
NOREVERSE	X	X	X
UNPLAN	X	X	X

Usage

The /RTAKEOVER command requests a remote takeover of an active IMS subsystem by a tracking subsystem at a secondary site. The remote takeover causes the tracking subsystem to finish processing and shut down.

If entered on an active subsystem, /RTAKEOVER also causes the active subsystem to shut down. Once the remote tracking subsystem has shut down and remote takeover has successfully completed, one or more subsystems may be brought up in an active role at the secondary site and started with standard IMS restart commands.

Unless you specify the NOREVERSE keyword, a remote takeover can be reversed, even after the /RTAKEOVER command has been issued. The NORTA parameter on the CHANGE.SG command can be used to reverse a remote takeover. See *IMS Version 9: Operations Guide*.

/RTAKEOVER for an Active Subsystem

/RTAKEOVER is issued on the active IMS subsystem to initiate a planned remote takeover and must be specified with either the FREEZE keyword or the DUMPQ keyword.

© Copyright IBM Corp. 1974, 2003

/RTAKEOVER FREEZE indicates that a /CHECKPOINT FREEZE type of shutdown is performed before the planned takeover occurs.

/RTAKEOVER DUMPQ indicates that a /CHECKPOINT DUMPQ type of shutdown is performed before the planned takeover occurs. This form of takeover provides that all relevant log records reach the tracking subsystem such that no data is lost. This form of takeover allows the capability to rebuild the message queues during the new active start.

/RTAKEOVER must be entered for at least one IMS subsystem in the global service group (GSG) at the active site for which takeover is to occur. The other IMS subsystems at the active site must also be shut down, either with the /RTAKEOVER command, or some form of the /CHECKPOINT command that shuts the system down.

Once the active sends all of its log data sets, the active IMS subsystem shuts down. When all of the active subsystems in the global service group have shut down, the tracking subsystem then completes tracking work, stops online forward recovery (OFR), changes the role of the service group (tracking to active) in the RECON data set, and shuts down.

/RTAKEOVER for a Tracking Subsystem

/RTAKEOVER UNPLAN is issued on the tracking IMS subsystem to initiate an unplanned remote takeover after the active site fails unexpectedly. /RTAKEOVER UNPLAN causes the tracking subsystem to complete tracking work, stops online forward recovery (OFR), changes the role of the service group (tracking to active) in the RECON data set, and shuts down.

Recommendation: Specify the NOREVERSE keyword to allow the tracking IMS subsystem to save and process all data it has received from the active site, regardless of whether that data was committed on the active IMS subsystem.

Although you can still reverse a remote takeover even if you specify NOREVERSE, you should not reverse it; in this case, you receive message DFS4122A when you restart the tracking subsystem.

If you do not specify NOREVERSE, the tracking IMS subsystem discards any uncommitted data it has received from the active subsystem.

Examples

Example for /RTAKEOVER Command at Active Site

Entry ET:

```
/RTA FREEZE
```

Response ET:

```
DFS2939I REMOTE SITE PLANNED TAKEOVER IN PROGRESS SYS3
DFS2719I MSDB CHECKPOINT WRITTEN TO MSDBCP2  SYS3 DFS994I
         *CHKPT 94308/160026**FREEZE*  SYS3
DFS3499I ACTIVE DDNAMES: MODBLKSA IMSACBB  FORMATA  MODSTAT ID:  2 SYS3
DFS3804I LATEST RESTART CHKPT: 94308/160026, LATEST BUILDQ CHKPT: 94308/154950 SYS3
DFS4036I CONVERSATION ENDING WITH SERVICE GROUP STLSITE2 SYS3
DFS3257I ONLINE LOG CLOSED ON DFSOLP01 SYS3
DFS2484I JOBNAME=JT160031 GENERATED BY LOG AUTOMATIC ARCHIVING SYS3
DFS092I  IMS LOG TERMINATED    SYS3
DFS4036I CONVERSATION ENDED  WITH SERVICE GROUP STLSITE2 SYS3
```

```
DFS4024I  STOP SERVGRP PROCESSING (INTERNAL) COMPLETE SYS3
DFS2091I IMS TIMER SERVICE SHUTDOWN COMPLETE SYS3
DFS0617I RDS BUFFERS HAVE BEEN SUCCESSFULLY PURGED SYS3
```

Response received at the Tracking system:

```
DFS2932I DATABASE UPDATES PRIOR TO SYSTEM SYS3 TAKEOVER HAVE BEEN
 ROUTED SYS3
```

Example for /RTAKEOVER DUMPQ at Active Site

Entry ET:

```
/RTA DUMPQ
```

Response ET:

```
DFS2939I REMOTE SITE PLANNED TAKEOVER IN PROGRESS SYS3
DFS2719I MSDB CHECKPOINT WRITTEN TO MSDBCP2  SYS3
DFS994I  *CHKPT 94308/165340**DUMPQ**  SYS3
DFS3499I ACTIVE DDNAMES: MODBLKSA IMSACBB  FORMATA  MODSTAT ID:  2 SYS3
DFS3804I LATEST RESTART CHKPT: 94308/165340, LATEST BUILDQ CHKPT: 94308/165340 SYS3
DFS4036I CONVERSATION ENDING WITH SERVICE GROUP STLSITE2 SYS3
DFS3257I ONLINE LOG CLOSED ON DFSOLP01 SYS3
DFS2484I JOBNAME=JT165345 GENERATED BY LOG AUTOMATIC ARCHIVING SYS3
DFS092I  IMS LOG TERMINATED    SYS3
DFS4036I CONVERSATION ENDED  WITH SERVICE GROUP STLSITE2 SYS3
DFS4024I  STOP SERVGRP PROCESSING (INTERNAL) COMPLETE SYS3
DFS2091I IMS TIMER SERVICE SHUTDOWN COMPLETE SYS3
DFS0617I RDS BUFFERS HAVE BEEN SUCCESSFULLY PURGED SYS3
```

Response received at the Tracking system:

```
DFS2932I DATABASE UPDATES PRIOR TO SYSTEM SYS3 TAKEOVER HAVE BEEN
ROUTED SYS3
```

Example for /RTAKEOVER UNPLAN at Remote Site

Entry ET:

```
/RTA UNPLAN
```

Response ET:

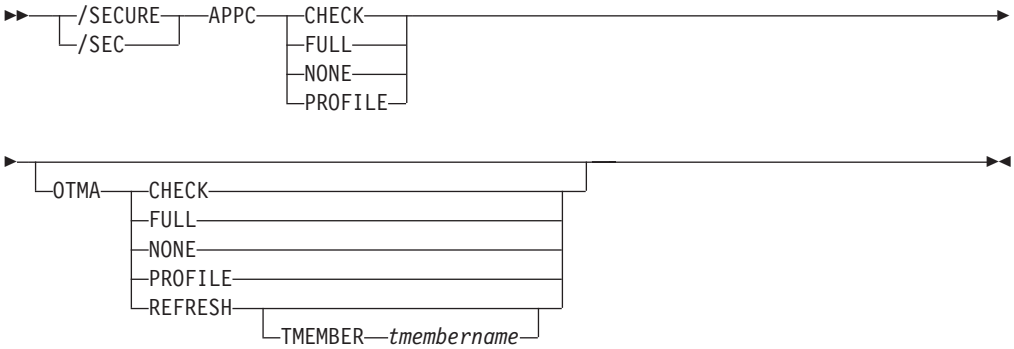
```
DFS4123I UNPLANNED TAKEOVER IN PROGRESS
DFS2913I CONVERSATION WITH IMS SYS3 TERMINATED: TRK SYS SHUTDOWN
DFS2913I CONVERSATION WITH IMS IMS2 TERMINATED: TRK SYS SHUTDOWN
DFS2500I DATASET  IMZ00007 SUCCESSFULLY DEALLOCATED
DFS2500I DATASET  IMZ00015 SUCCESSFULLY DEALLOCATED
DFS2934I LOG TRUNCATION STARTED FOR IMS: IMS2
DFS2934I LOG TRUNCATION STARTED FOR IMS: SYS3
DFS2943I THERE ARE NO DATA SETS TO TRUNCATE FOR SYS3
DFS2908I DATABASE UPDATES COMPLETE FOR REMOTE TAKEOVER PROCESSING
DFS2500I DATASET  IMZ00020 SUCCESSFULLY CREATED
DFS2500I DATASET  IMZ00021 SUCCESSFULLY ALLOCATED
DFS4040I SHADOW LOG CREATE, DD=IMZ00020, DSN=IMSTESTL.RSR.SLDS1.N0000016
DFS2500I DATASET  IMZ00021 SUCCESSFULLY DEALLOCATED
DFS2935I TRACKING LOG DATA SETS HAVE BEEN TRUNCATED AT 000000000000209C:
DFS2936I IMSTESTL.RSR.SLDS1.N0000012
DFS2500I DATASET  IMZ00016 SUCCESSFULLY DEALLOCATED
DFS2500I DATASET  IMZ00020 SUCCESSFULLY DELETED
DFS4126I TAKEOVER COMPLETE
DFS994I *CHKPT 94310/160240**FREEZE*
DFS3499I ACTIVE DDNAMES: MODBLKSA IMSACBA  FORMATA  MODSTAT ID:    3
DFS3804I LATEST RESTART CHKPT: 94310/160240, LATEST BUILDQ CHKPT: 94310/155301
DFS3257I ONLINE LOG CLOSED ON DFSOLP01
DFS2484I JOBNAME=JT160245 GENERATED BY LOG AUTOMATIC ARCHIVING
```

```
DFS092I IMS LOG TERMINATED
DFS2091I IMS TIMER SERVICE SHUTDOWN COMPLETE
DFS0617I RDS BUFFERS HAVE BEEN SUCCESSFULLY PURGED
```

Explanation: An unplanned takeover is successfully initiated for a tracking subsystem that was tracking 2 active subsystems (SYS3 and IMS2).

Chapter 53. /SECURE

Format



Environments and Keywords

Table 141 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 141. Valid Environments for the /SECURE Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/SECURE	X		X
APPC	X		X
OTMA	X		X

Usage

The /SECURE command is used to control the RACF security level. It is used for administrative control of the IMS environment and as an emergency operations control command to throttle RACF activity without requiring an IMS shutdown.

APPC
When used with the CHECK, FULL, NONE, or PROFILE parameters. APPC controls the RACF security level for input from LU 6.2 devices. The /DISPLAY APPC command can be used to show the security level that is currently in effect. At IMS startup, the security default is FULL.

CHECK
Causes existing RACF calls to be made. IMS commands are checked using the RACF resource class of CIMS. IMS transactions are checked using TIMS. Disables MVS System Authorization Facility security for IMS allocate PSBs (APSBs).

FULL
Causes the same processing as the CHECK parameter but uses additional RACF calls to create the security environment for dependent regions and enables MVS System Authorization Facility security for IMS APSBs for all CPI Communications driven application programs.

NONE
Does not call RACF within IMS for security verification. RACF security

© Copyright IBM Corp. 1974, 2003

verification in APPC/MVS is not affected. Disables MVS System Authorization Facility security for IMS APSBs.

PROFILE
Causes the values in the TP profile for each transaction to be used. If the TP profile is not defined for a transaction, or if the TP profile does not specify a RACF security level, then the default security is CHECK.

OTMA
Is used with the CHECK, FULL, NONE, or PROFILE parameters to control the RACF security level for input from IMS Open Transaction Manager Access (OTMA) clients. The `/DISPLAY OTMA` command can be used to show the security level that is currently in effect. After an IMS cold start, the security default is FULL if the IMS start-up parameter OTMASE= is not used. If the IMS initialization parameter OTMASE is not specified, IMS retains OTMA security settings (established by a `/SECURE OTMA` command) after a warm start or emergency restart. If the OTMASE parameter is used, the security option for OTMA will be determined by the OTMASE= setting.

Related Reading: For more information about the OTMASE parameter, see the *IMS Version 9: Installation Volume 2: System Definition and Tailoring*.

CHECK
Causes existing RACF calls to be made. IMS commands are checked using the RACF resource class of CIMS. IMS transactions are checked using TIMS.

FULL
Causes the same processing as the CHECK parameter but uses additional RACF calls to create the security environment for dependent regions.

NONE
Does not call RACF within IMS for security verification.

PROFILE
Causes the values in the Security Data section of the OTMA message prefix for each transaction to be used.

REFRESH
OTMA caches the ACEE for a userid to reduce the amount of RACF I/O. As a result, a refresh for the cached ACEE is needed after the RACF database is updated. Issuing the `/SEC OTMA REFRESH` command without the TMEMBER option will perform the ACEE refresh for all userids for all the OTMA clients. However, the actual ACEE refresh occurs when the next OTMA message for the userid is received. This is designed to prevent all the RACF ACEE refreshes from happening at one time.

Examples

Example 1 for /SECURE Command

Entry ET:

```
/DIS APPC
```

Response ET:

```
IMSLU     #APPC-CONV SECURITY STATUS   DESIRED
IMSLUNME           0 PROFILE  ENABLED  ENABLED
*91242/163820*
```

Explanation: Enter `/DISPLAY APPC` to see which security checking option is in effect.

Entry ET:

```
/SECURE APPC FULL
```

Response ET:

```
DFS058I SECURE COMMAND COMPLETED
```

Example 2 for /SECURE Command

Entry ET:

```
/DIS OTMA
```

Response ET:

```
GROUP/MEMBER      XCF-STATUS   USER-STATUS    SECURITY
HARRY
-APPL8            ACTIVE       SERVER         FULL
*95068/175730*
```

Explanation: Enter `/DISPLAY OTMA` to see which security checking option is in effect.

Entry ET:

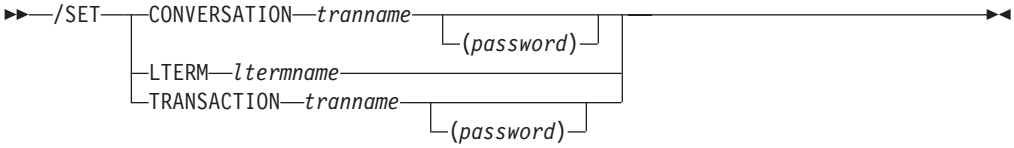
```
/SECURE OTMA FULL
```

Response ET:

```
DFS058I SECURE COMMAND COMPLETED
```

Chapter 54. /SET

Format



Environments and Keywords

Table 142 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 142. Valid Environments for the /SET Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/SET	X		X
CONVERSATION	X		X
LTERM	X		X
TRANSACTION	X		X

Usage

/SET establishes the destination of all messages entered into this terminal to another terminal or to a particular transaction code. If the terminal is in conversation, the /SET command also sets the destination of only the next message to the specified transaction.

The status fields of /DISPLAY LINE PTERM, /DISPLAY NODE, or /DISPLAY USER indicate if a physical terminal, node, or user is in preset destination mode (PRST), and display the destination transaction code or logical terminal name.

CONVERSATION
Directs the next input message to a selected transaction. The terminal being used must be in a conversation that is waiting for an input message. For example, the response message must have been received.

In an IMSplex, if global resource information is kept in Resource Manager, the /SET command sets a transaction for the next input message both globally and locally. If global resource information is not kept in Resource Manager, the /SET command sets the transaction just locally.

LTERM
Specifies the logical terminal that is the destination of all messages entered into this terminal.

The mode established by /SET LTERM is called preset mode. If the preset mode is established from a 3270 and user-defined formats are not being used, input message must be entered from a cleared screen. Preset mode can be reset by:

```
/IAM
/STOP LINE PTERM
/STOP NODE command
/STOP USER command
```

© Copyright IBM Corp. 1974, 2003

```
/RESET
/STOP NODE
/STOP LINE
/STOP USER
```

Once a destination is preset, the terminal operator cannot enter the destination (logical terminal name) as the first part of the message.

In a multiple systems configuration, the name of a remote logical terminal can be specified. If the preset destination is to be deleted (/RESET) or changed (/SET), the command must be entered from some other valid component.

TRANSACTION
Specifies the transaction code that is the destination of all messages entered into this terminal. The mode established by `/SET TRANSACTION` is called preset mode. If the preset mode is established from a 3270 and user-defined formats are not being used, input messages must be entered from a cleared screen. Preset mode can be reset by:

```
/IAM
/START LINE PTERM
/RESET
/STOP NODE
/STOP LINE
/STOP USER
```

Once a destination is preset, the terminal operator cannot enter the destination (transaction code) as the first part of the message. In a multiple systems configuration, the name of a remote transaction can be specified. The terminal cannot be in conversation.

Examples

Example 1 for /SET Command

Entry ET:

```
/SET CONVERSATION CONVTRAN(password)
```

Response ET:

```
DFS058I  SET COMMAND COMPLETED
```

Explanation: Any message entered from this terminal is sent to conversation CONVTRAN.

Example 2 for /SET Command

Entry ET:

```
/SET LTERM CNTRL
```

Response ET:

```
DFS058I  SET COMMAND COMPLETED
```

Explanation: Any message entered from this terminal is sent to LTERM CNTRL.

Example 3 for /SET Command

Entry ET:

```
/SET TRANSACTION IMS(password)
```

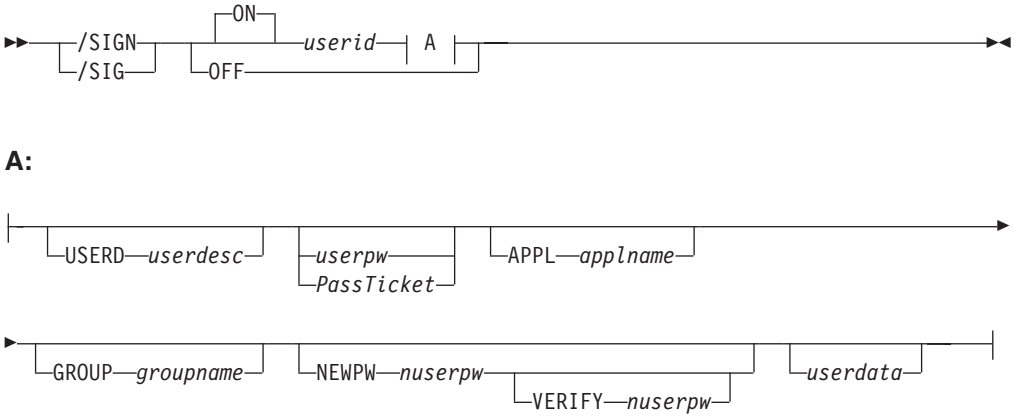
Response ET:

```
DFS058I  SET COMMAND COMPLETED
```

Explanation: Any message entered from this terminal is sent to transaction IMS.

Chapter 55. /SIGN

Format



Environments

Table 143 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command can be issued.

Table 143. Valid Environments for the /SIGN Command

Command	DB/DC	DBCTL	DCCTL
/SIGN	X		X

Usage

The /SIGN command is used to sign on and sign off at terminals attached to IMS. This command enables IMS to identify who is using the terminal and to determine if you are authorized to enter the transaction or command.

When SGN=G, Z, or M is specified, the user can sign on multiple times to both STATIC and ETO terminals when the structure name is different from the user ID.

The status fields of /DISPLAY NODE and /DISPLAY LINE PTERM indicate whether a terminal is signed on with the word SIGN.

ON

/SIGN ON must be issued for any physical terminal or user ID requiring a signon, or the transaction entered is rejected.

From terminals that require signon, commands other than /SIGN or /RCLSDST are rejected if transaction authorization is requested. Static terminals requiring a signon also have enhanced command authorization with RACF or an equivalent product if RCF=S or RCF=A is specified at system startup.

At terminals not requiring signon, transactions are passed to RACF, an equivalent security product, or a user exit for authorization checking. If /SIGN ON is entered at a terminal not requiring a signon, the signon is processed as if the terminal required a signon. That is, the terminal is placed in a signed on status

© Copyright IBM Corp. 1974, 2003

with the user ID until a /SIGN OFF or another /SIGN ON command is entered. For switched terminals, the /IAM command must be issued before the /SIGN ON command.

After any IMS restart or terminal disconnect, the remote terminal operator is required to sign on again using the /SIGN ON command. A terminal can be disconnected by:

- A switched line disconnect
- A VTAM CLSDST
- A line shutdown
- The /IDLE command
- Auto logoff

Signon status is also reset by the /START LINE, /START LINE PTERM, and /START NODE commands and auto signoff.

The remote terminal operator must wait at a static physical terminal for confidential responses, because responses queued for a given physical terminal are sent even if the physical terminal is signed off. If the remote terminal operator must be absent, the /LOCK command can be used to prevent output from being received. Confidential output sent to a dynamic user is queued to the user instead of to the physical terminal when the user has signed off. A successful signon of an existing user turns off the DEADQ status for the user, if that status exists.

APPL
A keyword that notifies IMS that the following character string should be the application name used by IMS when IMS makes the SAF call to verify the user. The default application name used by IMS is the IMSID. The IMSID can be overridden by the SAPPLID= parameter in the IMS PROCLIB member DFSDCxxx. If the signon specifies a PassTicket instead of a password, the APPL parameter should specify the application name used when the PassTicket was created. The creator of the PassTicket can specify any value to identify an IMS subsystem.

If RACF is used, APPL= should specify the name of the RACF PTKTDATA profile for IMS as defined to RACF by the creator of the PassTicket. If the name of the PTKTDATA profile is the same as the IMSID, the APPL keyword is not needed. For more information on the RACF secured signon PassTicket, see the *RACF Macros and Interfaces* manual. For more information on using the secured signon function, see the *RACF Security Administrator's Guide*.

GROUP
Is an optional keyword indicating a group name of 8 characters or fewer that is associated with the user ID.

NEWPW
Is an optional keyword indicating a new user password of 8 characters or fewer that replaces the current user password specified in *userpw*.

nuserpw
Is a new password of 8 characters or fewer that is associated with the user identification.

PassTicket
A one-time password that is generated by a requesting product or function.

The *PassTicket* is an alternative to the RACF password. Using a *PassTicket* removes the need to send RACF passwords across the network in clear text.

USERD
Is a user descriptor name. This user descriptor name is used in the sign on. The `userdesc` parameter must be a user ID, node name or DFSUSER.

userdata
Is user identification information that has been defined to IMS with the (RACF), equivalent security product or the user exit routine, DFSCSGN0. For RACF, this information consists of the following:

```
userpw  GROUP groupname   NEWPW nuserpw
```

userid
Is a user identification of 8 characters or fewer.

userpw
Is a password of 8 characters or fewer that is associated with the user identification.

VERIFY
Is an optional keyword that requests IMS to verify the new password entered. IMS verifies the new password before passing it to RACF or to the IMS signon exit routines. This keyword can also be used as an alternative to re-entering the password on the DFS3656 panel.

Restriction: You can use this keyword only when responding to an IMS `DFS3656A` message and as an alternative to re-entering the password on the DFS3656 panel.

For the user exit routine DFSCSGN0, the user ID and userdata parameter values are defined by the installation.

OFF
The `/SIGN OFF` command is used to complete a session on a terminal that required a signon. Static terminals in conversational mode cannot be signed off without first issuing an `/EXIT` or `/HOLD` command.

Another method of signing off a terminal is to reenter the `/SIGN ON` command. This method initiates a new signon at the terminal without having to enter the `/SIGN OFF` command.

The `/SIGN OFF` command resets status that is not significant such as preset mode, test mode, response mode, lock lterm, pstop lterm, and purge lterm.

`/SIGN OFF` for ETO users will also take other actions depending on the recovery settings for the user:

RCVYCONV=NO
`/SIGN OFF` causes any IMS conversations (active and held) for an ETO user to be terminated. Any conversational message that is queued or being processed has its output response message delivered asynchronously.

RCVYFP=NO
`/SIGN OFF` cause s Fast Path status and messages for an ETO user to be discarded.

If global resource information is kept in Resource Manager, `/SIGN OFF` deletes the user ID from Resource Manager (if single user signon enforced) and resets

status globally. If the user has no status, /SIGN OFF deletes the user and associated lterms from Resource Manager.

Examples

Example 1 for /SIGN Command

Entry ET:

```
DFS3649A /SIGN COMMAND REQUIRED FOR IMS

DATE: 11/03/92     TIME: 14:39:33

NODE NAME: DT327001

USERID: IMSUS01

PASSWORD: IMSPW01

USER DESCRIPTOR:
GROUP NAME:
NEW PASSWORD:

   OUTPUT SECURITY AVAILABLE
```

Response ET:

```
DFS3650I SESSION STATUS FOR IMS

DATE: 11/03/92     TIME: 14:41:48
NODE NAME:           DT327001
USERID:              IMSUS01
PRESET DESTINATION:

CURRENT SESSION STATUS:

   OUTPUT SECURITY AVAILABLE
```

Explanation: The user with user ID IMSUS01 and password IMSPW01 has successfully signed on to a dynamic terminal. The signon is done with the panel (DFS3649A).

Example 2 for /SIGN Command

Entry ET:

```
/SIGN IMSUS02 IMSPW02
```

Response ET:

```
DFS3650I SESSION STATUS FOR IMS

DATE: 11/03/92     TIME: 14:41:48
NODE NAME:           DT327001
USERID:              IMSUS02
PRESET DESTINATION:

CURRENT SESSION STATUS:

   OUTPUT SECURITY AVAILABLE
```

Explanation: The user with user ID IMSUS02 and password IMSPW02 has successfully signed on to a dynamic terminal. The signon is done with the /SIGN command.

Example 3 for /SIGN Command

Entry ET:

```
/SIGN IMSUS03 IMSPW03
```

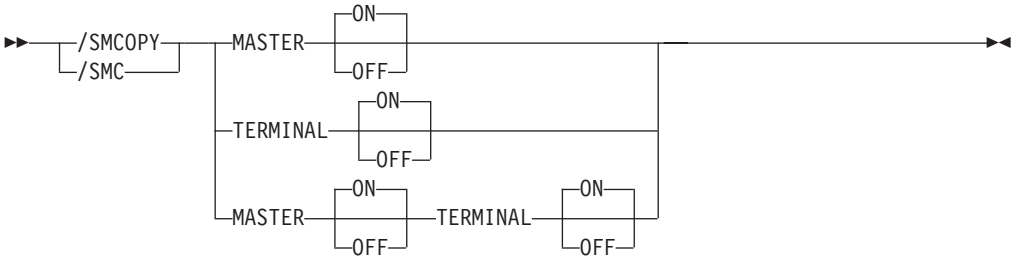
Response ET:

```
 DFS3650I SESSION STATUS FOR IMS

 DATE: 11/03/92     TIME: 14:45:53
 NODE NAME:           L3270A
 USERID:              IMSUS03
 PRESET DESTINATION:

 CURRENT SESSION STATUS:

 NO OUTPUT SECURITY AVAILABLE
```

Explanation: The user with user ID IMSUS03 and password IMSPW03 has successfully signed on to a static terminal.

Chapter 56. /SMCOPY

Format



Environments and Keywords

Table 144 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 144. Valid Environments for the /SMCOPY Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/SMCOPY	X		X
MASTER	X		X
TERMINAL	X		X

Usage

The /SMCOPY command is used by the master terminal operator to control the printing of certain output to the secondary master terminal.

Table 145 shows the IMS commands that can be printed on the secondary master terminal.

Table 145. Commands that Are Logged to the Secondary Master Terminal

/ACTIVATE	/ALLOCATE
/ASSIGN	/CHECKPOINT
/CLSDST	/COMPT
/DBDUMP	/DBRECOVERY
/DELETE	/DEQUEUE
/DISPLAY	/IDLE
/MODIFY	/MONITOR
/MSASSIGN	/OPNDST
/PSTOP	/PURGE
/QUIESCE	/RCLSDST
/RCOMPT	/RMCHANGE
/RMDELETE	/RMGENJCL
/RMINIT	/RMLIST
/RMNOTIFY	/RSTART
/SECURE	/START
/STOP	/SWITCH
/TRACE	/UNLOCK SYSTEM

© Copyright IBM Corp. 1974, 2003

System definition establishes whether the commands and their responses will be printed on the secondary master and the origin of the printed command (master terminal, remote terminal, or both). /SMCOPY provides online control of the printing established by system definition.

MASTER
Turns ON or OFF the printing of the above subset of IMS commands and command responses when issued from the master terminal.

TERMINAL
Turns ON or OFF the printing of the above subset of IMS commands and command responses when issued from terminals other than the master terminal.

Input coming through Operations Manager is not reflected in the secondary master. This also applies to input coming from MCS/E-MCS terminals.

Example for /SMCOPY Command

A system definition has established that copies of the above subset of commands and command responses, when issued from any terminal, will be printed on the secondary master terminal (operand COPYLOG=ALL was specified on the COMM macro).

Entry ET:

```
/SMCOPY TERMINAL OFF
```

Response ET:

```
DFS058I  SMCOPY COMMAND COMPLETED
```

Explanation: The secondary master terminal does not receive copies of IMS commands and command responses issued from remote terminals. Commands and responses issued from the master terminal are still received.

Chapter 57. /SSR

Format

Environments

Table 146 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command can be issued.

Table 146. Valid Environments for the /SSR Command

Command	DB/DC	DBCTL	DCCTL
/SSR	X	X	X

Usage

/SSR is a multisegment command that allows the IMS operator to enter an external subsystem command as part of the command input. (The external system is not a CCTL subsystem.) Routing is the only function IMS performs. The command is processed by the external subsystem and the response (from the external subsystem, not CCTL) is routed back to the entering terminal.

All /SSR formats require an EOM indication to denote end-of-message. An EOS indication must be included for all segments that precede the last segment.

text is the alphanumeric external subsystem command.

Example for /SSR Command

Entry ET:

```
/SSR ;START DATABASE (DSN8D22P)
```

Response ET:

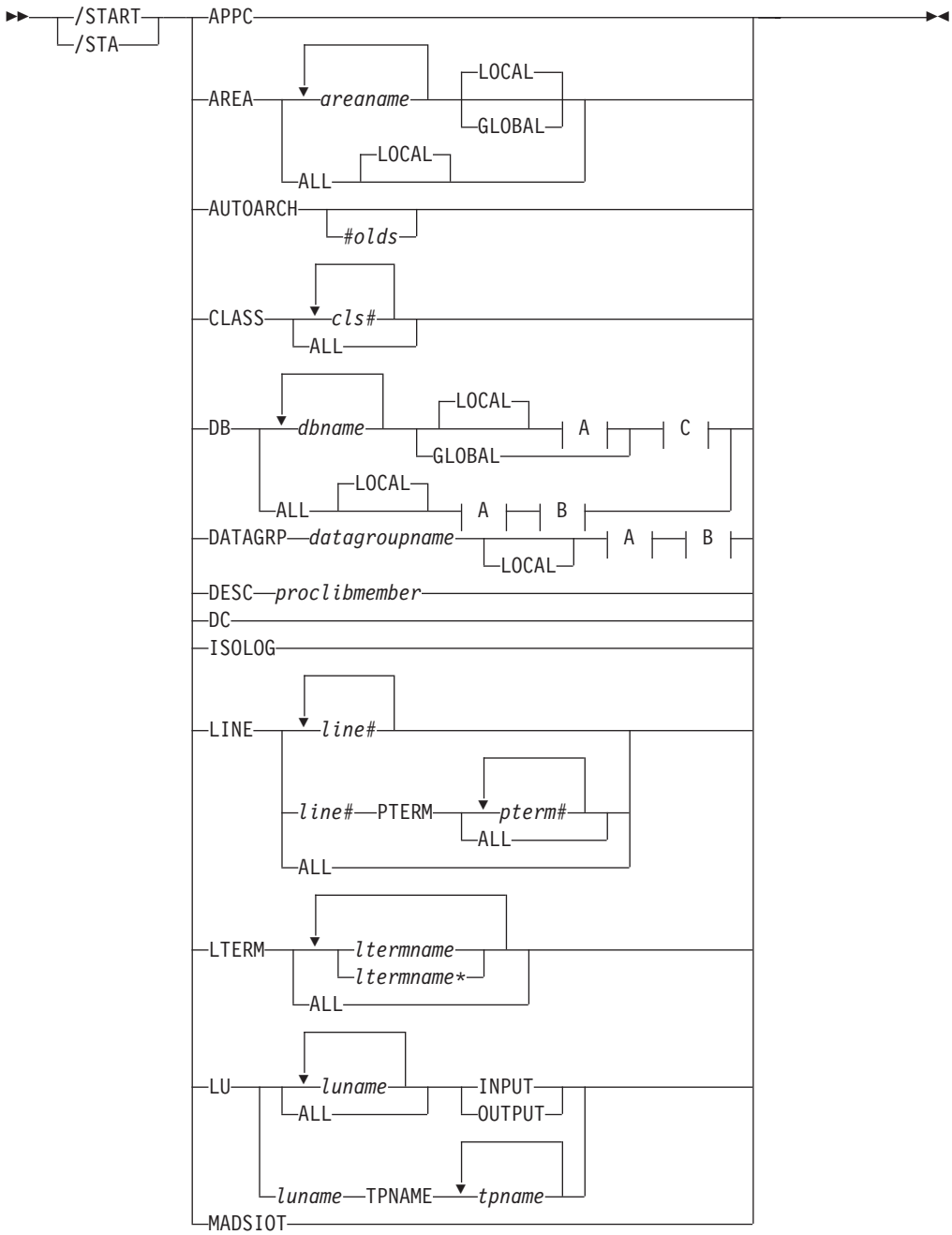
```
DFS058I  SSR COMMAND COMPLETED

DSN9022I ; DSNTDDIS 'START DATABASE' NORMAL COMPLETION

Explanation:  The START DATABASE command is successfully routed
to the DB2 subsystem for processing.
```

© Copyright IBM Corp. 1974, 2003

Chapter 58. /START

Format

/START Command: APPC through MADSIOT

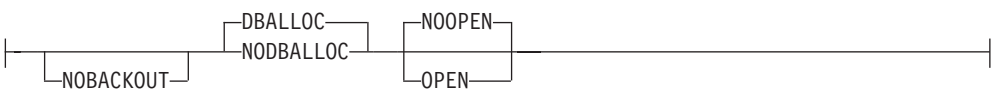


A:

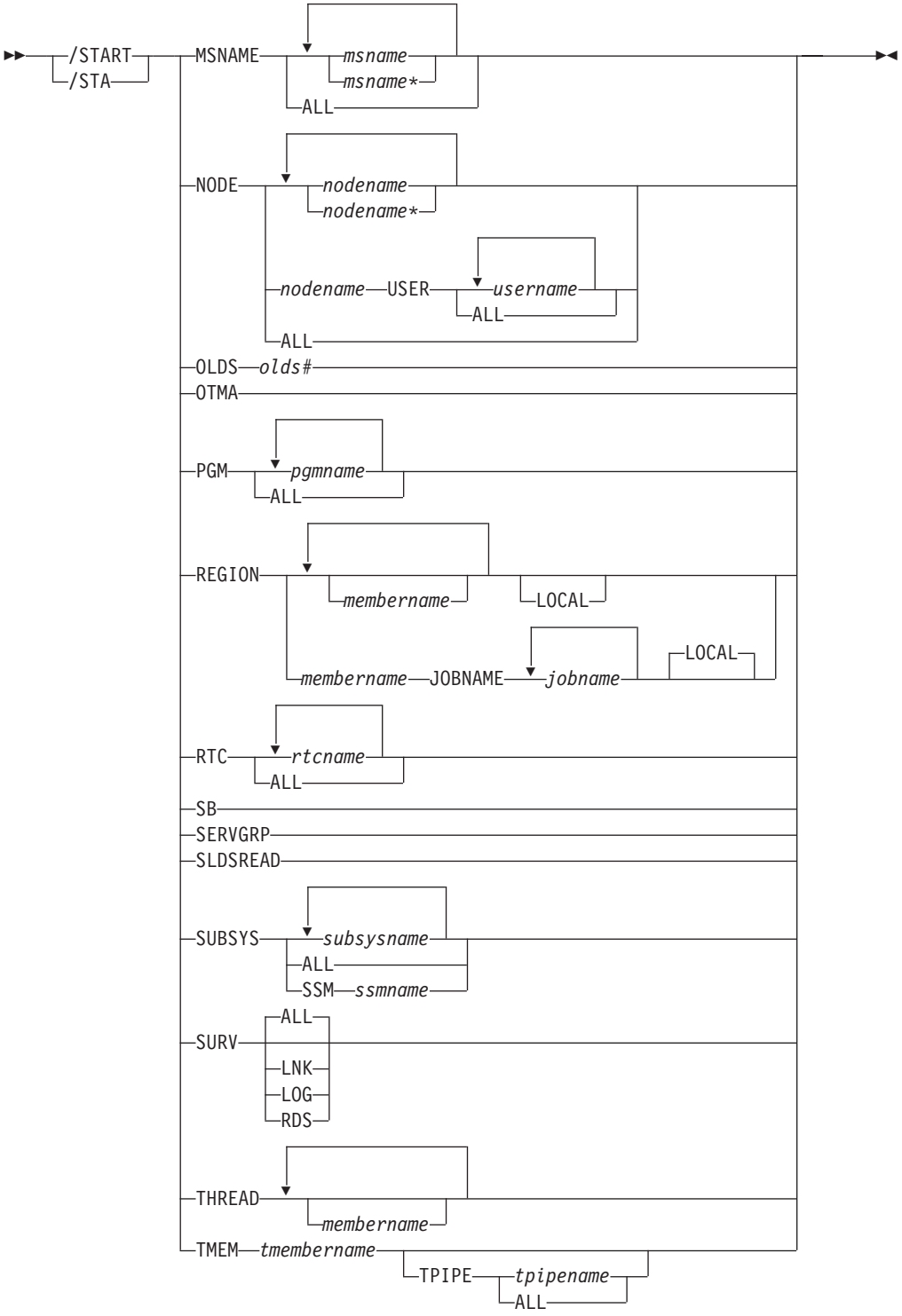
© Copyright IBM Corp. 1974, 2003

B:

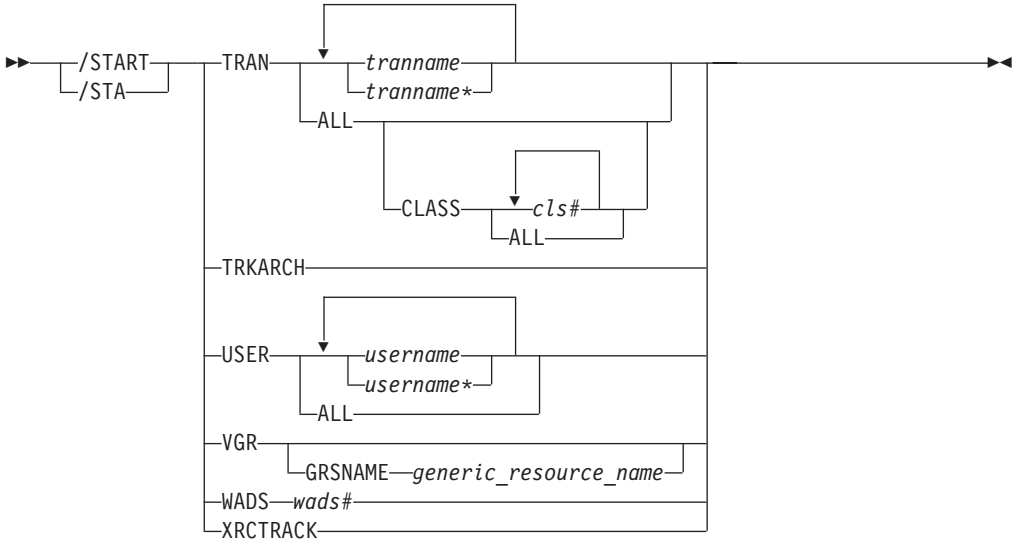
C:



/START Command: MSNAME through TMEM



/START Command: TRAN through XRCTRACK



Environments and Keywords

Table 147 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 147. Valid Environments for the /START Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/START	X	X	X
ACCESS	X	X	
APPC	X		X
AREA	X	X	
AUTOARCH	X	X	X
CLASS	X		X
DB	X	X	
DATAGRP	X	X	
DBALLOC	X	X	
DC	X		X
DESC	X		X
GLOBAL	X	X	
GRSNAME	X		X
INPUT	X		X
ISOLOG	X	X	X
JOBNAME	X	X	X
LINE	X		X
LOCAL	X	X	
LTERM	X		X
LU	X		X
MADSIOT	X	X	
MSNAME	X		X

Table 147. Valid Environments for the /START Command and Keywords (continued)

Command / Keywords	DB/DC	DBCTL	DCCTL
NOBACKOUT	X	X	
NODBALLOC	X	X	
NODE	X		X
NOOPEN	X	X	
OLDS	X	X	X
OPEN	X	X	
OTMA	X		X
OUTPUT	X		X
PGM	X	X	X
PTERM	X		X
REGION	X	X	X
RTC	X		X
SB	X	X	
SERVGRP	X	X	X
SLDSREAD	X	X	X
SSM	X	X	X
SUBSYS	X	X	X
SURV	X		X
THREAD	X	X	
TMEM	X		X
TPIPE	X		X
TPNAME	X		X
TRAN	X		X
TRKARCH	X	X	X
USER	X		X
VGR	X		X
WADS	X	X	X
XRCTRACK	X	X	X

Usage

The /START command makes IMS resources available for reference and use.

The /START command can be used on HALDBs. For more information see Appendix H, “High Availability Large Database Commands,” on page 867

/START also checks the validity of all parameters entered by the terminal operator. If an error is detected on parameters that are independent of one another, only the invalid parameters are indicated as being in error and the /START command processes the rest of the parameters. For example,

```
/START LINE 4 6 200
DFS058 START COMMAND COMPLETED EXCEPT LINE 200
```

signifies parameter value 200 is not a valid line number.

When a resource becomes available, the system parameters used for this initialization of IMS are displayed in message DFS1929I. The system parameters are also written to the job log.

APPC
Instructs IMS to activate the connection to APPC/z/OS and to start accepting transaction schedule requests from APPC/z/OS. This command reverses the effect of a /PURGE APPC command or a /STOP APPC(CANCEL) command.

The /START APPC command sets the desired status to ENABLED. The current status is initially set to STARTING. When APPC/z/OS responds to the start request, the status changes to either ENABLED or FAILED.

AREA
Specifies the specific areas of DEDBs to be allocated. For z/OS, /START AREA can be used to reallocate DEDB areas.

For areas on an RSR tracking subsystem, /START AREA is used to resume tracking for those areas that were stopped by a previous /DBRECOVERY command or by errors found during tracking subsystem processing. /START AREA also starts online forward recovery (OFR) for those areas that are not current with mainline tracking.

For virtual storage option (VSO) areas that have been defined with the PREOPEN option, /START AREA causes the areas to be preopened. If the VSO area is defined with the PRELOAD option, /START AREA causes the area to be opened and loaded into the z/OS data space.

Restriction: This command only applies to the IMS subsystem on which it is entered; it does not preload or preopen areas on other IMS subsystems in the sysplex that share the area.

The /START AREA command has no effect on VSO areas that are in virtual storage when the command is issued.

The output of the /START AREA command is changed when the command is entered through the OM API. In this case, the DFS058I message is not returned to OM. For commands that specify GLOBAL, only the command master returns the asynchronous messages to OM. When a command is processed with the LOCAL keyword, all IMSs are able to return the asynchronous messages to OM. The command response returned to OM contains one or more of the following messages as appropriate.

Fast Path messages: DFS0011I, DFS140I, DFS0488I, DFS0666I, DFS1407I, DFS2980E, DFS2981E, DFS3320I, DFS3325I, DFS3342I, DFS3720I, DFS3824I

GLOBAL
The GLOBAL keyword applies when an IRLM is active. GLOBAL specifies that the command applies to all subsystems sharing the area. The GLOBAL keyword and the ALL parameter are mutually exclusive. The /START command is rejected if both ALL and GLOBAL are specified. The GLOBAL keyword requires that IRLM be active. The command will be rejected if IRLM is not active.

The GLOBAL keyword is not supported on an RSR tracking subsystem.

3 If the GLOBAL keyword on a command is entered from an OM API, the
3 command is processed only by the command master IMS. The command
3 master IMS will make DBRC calls to update the RECON with GLOBAL
3 status. It will also request IRLM NOTIFY to route and process the command

on sharing IMS systems, and then process the command locally. All other non-master IMSs ignore the /START command with the GLOBAL keyword.

Messages produced on the NOTIFIED systems will appear only on the system console and will not be routed back to the OM API which originally entered the command.

If multiple IMS systems have been explicitly specified in the route list, the master IMS system will process the command as described above. However, the non-master IMS systems, to which OM routes the command, will reject the command with the following return and reason code listed in Table 148:

Table 148. Return and reason code for GLOBAL keyword issued from the OM API

Return Code	Reason Code	Meaning
X'00000004'	X'00001000'	The command contained the GLOBAL keyword and was routed to more than one IMS system in the IMSPLEX. The non-master IMS systems will reject this command when OM routes the command to them. The master IMS system will process this command and use IRLM NOTIFY to route and process the command on the non-master IMS systems. See the discussion under the GLOBAL keyword.

LOCAL
Specifies that the command only applies to the IMS subsystem in which the command is entered. This command does not affect any other subsystem sharing the area.

LOCAL is the default.

/START AREA ALL causes message DFS0488 to be issued for every area that is not started successfully, but you do not see a DFS0488 message for every area that does start successfully. You do see a final DFS0488 message which indicates the end of command processing.

AUTOARCH
Is used to set the value to change the automatic archiving option selected at system initialization or to set the value to start automatic archiving after a previous /STOP AUTOARCH command. #olds is the number of OLDS that are to be filled before the /DBRC GENJCL ARCHIVE command is to be generated. It is optional and defaults to either the value specified at system initialization or to one. If /DBR NOFEOV or /DBD NOFEOV is issued before nn OLDS are filled, the number of OLDS currently filled will be archived.

/START AUTOARCH will not be carried over a warm start if a checkpoint has occurred after the command has been successfully completed, and before a warm start has been issued.

CLASS
Specifies transaction class, allowing scheduling of application programs to begin. Message regions must have appropriate classes assigned to them before scheduling will proceed.

DATABASE
Specifies the DBD name. The /START DATABASE command permits access from transactions that read or update databases. /START DATABASE can be used to

allocate or reallocate all databases other than DEDBs. An AREA command must be entered to allocate or deallocate DEDB AREAs.

For a DEDB, the /START DATABASE command also causes any unloaded randomizer, specified in the DBD source, to be reloaded.

When the name that is specified is for a partition, the action taken to allocate data sets varies. The action varies depending on the status of the master database and whether the DMB for the master database is already loaded. If a /DBRECOVERY command has not been issued against the master database, the DMB is already loaded in the following situations:

- The database is defined in SYSGEN as RESIDENT.
- One of the database partitions was previously accessed while this control region is running.

PHDAM or PHIDAM partitions that had the /DBRECOVERY command issued against them, cannot be allocated with the /START DATABASE DBALLOC command. However, if the partition databases have EEQEs, or the OPEN keyword is used with the /START DATABASE command, or the databases were previously authorized but not allocated, the partitions can be allocated. If you do not use the OPEN keyword, the PSINDEX partition is allocated and the PHDAM and PHIDAM partitions are not allocated until they are authorized.

When a /START DATABASE command is issued for all transactions whose processing program has access to a successfully started database, the USTOPPED attribute will be reset and any messages on the suspend queue for that transaction will be transferred to the normal queue.

If one or more of the named databases requires backout or recovery, and the database is registered in DBRC, the database requiring backout or recovery is dropped from the command and the remainder of the databases continue processing. If the database is not registered in DBRC, specify the NOBACKOUT keyword to inform IMS that it does not have to attempt to execute the failed backout again.

To start a HIDAM database, both the index and the data area DBD names must be specified. If a backout failure occurred for this database, the /START command causes the backout to be attempted again.

If the database specified in the command is being used by a batch message processing region, an error message is returned to the master terminal, and the command is ignored for the database named in the message. Processing continues for the other databases specified in the command. The master terminal operator must wait until the batch message processing concludes before reentering the command.

For databases on an RSR tracking subsystem, /START DATABASE is used to resume tracking for those databases that were stopped by a tracking subsystem processing. /START DATABASE also starts online forward recovery (OFR) for those databases that are not current with mainline tracking.

/START DATABASE ALL causes message DFS0488I to be issued. All databases that were defined at sysgen will be started if possible.

The output of the /START DATABASE command is changed when the command is entered through the OM API. In this case, the DFS058I message is not returned

to OM. The command response returned to OM contains one or more of the following messages as appropriate to the database type and the command completion.

Full Function Database messages: DFS030I, DFS132, DFS160, DFS216, DFS0402I, DFS0488I, DFS0740I, DFS1407, DFS2026, DFS3317I, DFS3318I, DFS3320I, DFS3325I, DFS3465I, DFS3466I

Fast Path Database messages: DFS140I, DFS666, DFS3062

For the results of issuing this command on a shared secondary index, see Appendix D, “Shared Secondary Index Database Commands,” on page 849

ACCESS

Specifies the type of access intended for the named database. This keyword overrides the database access specified at system definition. Changing the ACCESS parameter of a DEDB is allowed only when all the AREAS in the DEDB are not authorized by the subsystem.

Changing the access intent of a database with the /START DB ACCESS= command causes any VSO areas of the database to be removed from virtual storage. Because the access intent of a DEDB cannot be changed while any of its areas are authorized to an IMS subsystem, IMS closes any open areas before processing the command. If a VSO area is closed as a result of the command, it is also removed from the data space.

The /START AREA command must be used to reactivate the VSO options (VSO and PREOPEN) and PRELOAD option for the area. If an area is opened as a result of an access request rather than by the /START AREA command, it is opened as a non-VSO area.

Non-VSO areas with the PREOPEN option are closed as a result of the /START DB ACCESS= command. These areas are reopened either at the next access request for the area or by the /START AREA command.

The GLOBAL and ACCESS keywords are mutually exclusive. The /START command is rejected if both keywords are specified.

The meanings of the ACCESS parameter values are:

RO specifies that the named database is available for read-only processing on this IMS subsystem. The only programs that can use the database on this subsystem are those that have a PCB processing option of GO (PROCOPT=GO). Programs that access the data using the GO processing option might see uncommitted data, since a sharing IMS subsystem could be updating the database, which is opened for input only.

RD specifies that the named database is available for read-only processing in this IMS subsystem. Programs with update intent can be scheduled, but cannot update the database. ACCESS of RD differs from ACCESS of RO in that the data is read with integrity (locking is performed) and all programs can access the data, not just those with a processing option of GO. The database is opened for read only.

UP specifies that the named database is for update as well as read processing in the IMS subsystem.

EX specifies that the named database is to be used exclusively by this IMS subsystem. This exclusive access is guaranteed only when the database is registered to DBRC.

DBALLOC
Indicates that the databases within the data group are to be allocated. DBALLOC is the default except for /START DATABASE ALL commands.

NODBALLOC
Indicates that the databases within the data group are not to be allocated. This is the default for /START DATAGROUP. The databases will be allocated when they are scheduled. NODBALLOC is the default for /START DATABASE ALL commands.

GLOBAL
Specifies that the command applies to all subsystems sharing the database. GLOBAL requires that IRLM be active. The command will be rejected if IRLM is not active. The GLOBAL keyword and the ALL parameter are mutually exclusive. The /START command is rejected if both ALL and GLOBAL are specified.

The GLOBAL keyword is not supported on an RSR tracking subsystem.

If the GLOBAL keyword on a command is entered from an OM API, the command should only be routed to one IMS system in the IMSplex. The IMS that receives the command from OM will make DBRC calls to update the RECON with GLOBAL status. It will also request IRLM NOTIFY to route and process the command on sharing IMS systems, and then process the command locally.

Messages produced on the NOTIFIED systems will appear only on the system console and will not be routed back to the OM API which originally entered the command.

If multiple IMS systems have been explicitly specified in the route list, the master IMS system will process the command as described above. However, the non-master IMS systems, to which OM routes the command, will reject the command with the following return and reason code listed in Table 149:

Table 149. Return and reason code for GLOBAL keyword issued from the OM API

Return Code	Reason Code	Meaning
00000004	00001000	The command contained the GLOBAL keyword and was routed to more than one IMS system in the IMSPLEX. The non-master IMS systems will reject this command when OM routes the command to them. The master IMS system will process this command and use IRLM NOTIFY to route and process the command on the non-master IMS systems. See the discussion under the GLOBAL keyword.

LOCAL
Specifies that the command only applies to the IMS subsystem in which the command is entered. This command does not affect any other subsystem sharing the database.

LOCAL is the default.

NOBACKOUT

Suppresses backout restart for a database not registered in DBRC. If there was a prior dynamic backout or emergency restart backout failure, then a /START, command will attempt to perform the backout again. However, if the log data required to perform the backout has been archived, the backout must be performed by executing the batch backout utility.

If the database is registered in DBRC and is using share control, then DBRC is informed when batch backout is successfully executed, and the failing backout will not be attempted again when the /START command is issued.

If the database is registered in DBRC and is using recovery control, DBRC is not informed when batch backout is successfully executed. You must specify the NOBACKOUT keyword to inform IMS that it does not have to attempt to execute the failed backout again.

NOBACKOUT is not valid with the ALL parameter.

OPEN | NOOPEN

Indicates that the named databases should be opened or should not be opened as part of the /START DB processing. NOOPEN is the default except when the database has EEQEs or the database was previously authorized but not allocated. If the database has EEQEs or was previously authorized but not allocated, then the database will be opened until the NOOPEN keyword is specified. Operators need to be aware of the results of issuing this command. Check the return code in message DFS0488I.

Restrictions: The OPEN parameter is not supported:

- On a HALDB master
- In an RSR environment
- On an XRF alternate
- With keywords DATAGROUP, ALL, GLOBAL, or NOALLOC.

DATAGROUP

Specifies groups of DL/I databases and Fast Path DEDBs to be allocated. *Data groups* are logical groupings of databases and areas; they allow simplified command processing for databases. You define a data group in the RECON data set using the INIT.DBDSGRP command with parameters GRPNAME and DBGRP. DATAGROUP is valid on active and RSR tracking subsystems.

Related Reading: See “Group Parameters” on page 20 for more information on defining data groups.

For databases and areas on an RSR tracking subsystem, /START DATAGROUP is used to resume tracking for those areas that were stopped by a previous /DBRECOVERY command or by errors found during tracking subsystem processing. /START DATAGROUP also starts online forward recovery (OFR) for those databases and areas that are not current with mainline tracking.

After processing for a /START DATAGROUP completes, a DFS0488 message is issued indicating the end of processing. A DFS0488 message is also issued for every database or area that does not start successfully.

If the datagroup contains both full function and Fast Path databases, a DFS0488 message might be issued indicating the /START DATAGROUP command completed successfully before any messages are issued that indicate a Fast Path area did not start successfully. This situation is due to the asynchronous processing of Fast Path databases.

For virtual storage option (VSO) areas that have been defined with the PREOPEN option, /START AREA causes the areas to be preopened. If the VSO area is defined with the PRELOAD option, /START AREA causes the area to be opened and loaded into the z/OS data space.

Recommendation: Although you can use DBDS groups as well as database groups for this command, you should use database groups whenever possible to eliminate the overhead of converting the DBDS group to a database group.

ACCESS

Overrides the access intent for a database. The original database access is specified during IMS system definition.

Changing the access intent for a DEDB is allowed only when no PSBs are scheduled that access any areas in the DEDB. You might have to stop PSBs and regions that have wait-for-input (WFI) transactions scheduled before you can change a DEDB's access intent.

Changing the access intent of a database with the /START DB ACCESS= command causes any VSO areas of the database to be removed from virtual storage. Because the access intent of a DEDB cannot be changed while any of its areas are authorized to an IMS subsystem, IMS closes any open areas before processing the command. If a VSO area is closed as a result of the command, it is also removed from the data space.

The /START AREA command must be used to reactivate the VSO options (VSO and PREOPEN) and PRELOAD option for the area. If an area is opened as a result of an access request rather than by the /START AREA command, it is opened as a non-VSO area.

Non-VSO areas with the PREOPEN option are closed as a result of the /START DB ACCESS= command. These areas are reopened either at the next access request for the area or by the /START AREA command.

The GLOBAL and ACCESS keywords are mutually exclusive. The /START command is rejected if both keywords are specified.

The meanings of the ACCESS parameter values are:

RO specifies that the named database is available for read-only processing on this IMS subsystem.

The only programs which can use the database on this subsystem are those which have a PCB processing option of GO (PROCOPT=GO). Programs which access the data using the GO processing option might see uncommitted data, since a sharing IMS subsystem could be updating the database, which is opened for input only.

RD specifies that the named database is available for read-only processing on this IMS subsystem.

Programs with update intent can be scheduled, but cannot update the database. ACCESS=RD differs from ACCESS=RO in that the data is read with integrity (locking is performed) and all programs can access the data, not just those with a processing option of GO. The database is opened for read only.

UP specifies that the named database is for update as well as read processing in the IMS subsystem.

EX specifies that the named database is to be used exclusively by this IMS subsystem.

This exclusive access is guaranteed only when the database is registered to DBRC.

DBALLOC
Indicates that the databases within the data group are to be allocated. NODBALLOC is the default for a /START DATAGROUP command.

LOCAL
Specifies that the /START command only applies to the IMS subsystem in which the command is entered.

NODBALLOC
Indicates that the databases within the data group are not to be allocated. This is the default for START DATAGROUP. The databases will be allocated when they are scheduled. This command does not affect any other subsystem sharing the database.

LOCAL is the default.

DC
Opens the VTAM ACBs (if MNPS for XRF is used, then both the MNPS and APPLID ACBs are opened) if they are not already open, enables logons to IMS, and enables the following optional transaction manager functions:

- IMS generic resource support. The defined VTAM generic resource group is joined with GRSNAME in the IMS or DCC PROCLIB members.
- IMS persistent sessions support for RNR. Session activity that was suspended due to a major outage is resumed or terminated, as appropriate, if the RNR option was specified in the DFSDCxxx PROCLIB member.

If the /START DC command is issued on an XRF alternate system that is using MNPS, the command will only open the APPLID ACB. The MNPS ACB is not opened until XRF takeover processing.

DESCRIPTOR
Defines the LU62 descriptors from DFS62xxx PROCLIB member to IMS. The full PROCLIB member name needs to be specified. The PROCLIB member must start with DFS62.

ISOLOG
Indicates that the RSR tracking subsystem is to initiate a request for isolated log data from the isolated log sender at the active site. /START ISOLOG can only be entered on an RSR tracking subsystem. /START ISOLOG may be needed when:

- The active system failed and was not restarted, and the OLDS at the time of failure was closed using the log recovery utility.
- Batch jobs or utilities running at the active were unable to send logs.
- Permanent error status is shown for gaps in the output of a /DISPLAY TRACKING STATUS command. If the log problem at the active has been corrected, use /START ISOLOG to initiate retry.

Successful completion of syntax checking of the /START ISOLOG command results in the DFS058 START COMMAND COMPLETED message, although processing of the command continues asynchronously.

LINE
Makes communication lines that are idle and in a stopped or process stopped state available for use. It also terminates any conversations that are active on the line. All terminals are removed from looptest mode, MFSTEST mode, or

exclusive mode, and any terminals in signon status are reset. The /START LINE command without the PTERM keyword enables the line again (resulting in a BTAM LOPEN macro).

If the line must be enabled again, /START LINE must be entered prior to any /START LINE PTERM command.

The /START LINE PTERM command makes one or more physical terminals available for use. The physical terminals are removed from response mode, test mode, looptest mode, MFSTEST mode, preset destination mode, or exclusive mode, and resets terminals in signon status. If IMS encounters a stopped and idle line when processing /START LINE PTERM, it restarts the line. Any inoperable components are marked as operable.

To activate I/O operations for a LINE, LINE PTERM, or NODE without altering the status of the associated/specified terminals, use /RSTART instead of /START.

If an error is detected on parameters that are independent of one another, only the invalid parameters are indicated as being in error and processing continues for the rest of the parameters. This happens for /START LINE if:

- The specified line is already started or is not idle.
- Any terminals on the line had conversations that could not be canceled (because an application program is scheduled).
- The specified line could not be started because of IMS internal processing.
- The DD statement is missing from the IMS execution JCL.

/START LINE no longer resets preset mode, test mode, and response mode since these statuses are no longer significant and therefore are not kept after a /START LINE or restart command.

LTERM

Specifies the logical terminals to be started and resets the QLOCK state (QLOCK indicates that the LTERM is locked from sending any further output or from receiving input that can create additional output for the same LTERM until the state is reset by a specific request received on the session.) /START LTERM is rejected for remote logical terminals.

The LTERM keyword is only effective for existing LTERMs.

The LTERM parameter can be generic where the generic parameter specifies LTERMs that already exist.

If global resource information is kept in Resource Manager, the/START LTERM command allows messages to be queued to the LTERM from anywhere in the IMSplex and the change is reflected both in Resource Manager and in the local IMS system.

LUNAME TPNAME

Specifies the LU name that is to be started.

Specifying the keyword INPUT starts an luname for any input and synchronous outbound activities. Specifying the parameter ALL with the keyword INPUT causes all future LU 6.2 inbound and synchronous output activities to be started as well.

Specifying the keyword OUTPUT starts an luname for asynchronous outbound activities. Specifying the parameter ALL with the keyword OUTPUT causes all future LU 6.2 outbound asynchronous activities to be started as well.

Specifying neither INPUT nor OUTPUT is the same as specifying both INPUT and OUTPUT. The LU name is started for any input and both synchronous and

asynchronous outbound activities. Specifying the parameter ALL in this case also causes the start of all future LU 6.2 inbound activities, outbound synchronous, and asynchronous activities.

A network-qualified LU name is optional for the LUNAME keyword. If the LU name is not network-qualified and no TP name is specified, all the network-qualified LU names whose LU name matches the LU name specified are also started.

/START LUNAME TPNAME starts a particular tpname of an luname. The keyword OUTPUT is the default for this command.

If the specified resource does not exist, a structure is created to retain the status.

MADSIOT

Specifies the MADS I/O timing function. The /START MADSIOT command is valid only after the long busy handling function is disabled for a link failure, a structure failure, or a rebuild failure. When the /START MADSIOT command completes normally, one of the two following messages is returned to the operator's console:

- DFS1728E START MADSIOT COMMAND FAILED RSN=rrr
- DFS1727I MADSIOT TIMING FUNCTION RESUMED SUCCESSFULLY

The purpose of this command is to enable MADS I/O Timing function. If MADS I/O Timing list structure is not defined in DFSVSMxx, the command will be rejected. If MADS I/O Timing function is already enabled, the command will be ignored. If MADS I/O Timing function is not enabled and all sharing partners successfully connect to MADS I/O Timing list structure on the coupling facility, the command will complete successfully; if any sharing partners fails to connect to MASD I/O Timing list structure, the command will fail.

The output of the /STA MADSIOT command is changed when the command is entered through the OM API. In this case, the DFS058I message is not returned to OM. The command response returned to OM contains one or more of the following messages as appropriate.

Fast Path messages: DFS0023I, DFS0007I, DFS1270I, DFS1727I, DFS1552A, DFS1728E

MSNAME

Specifies the logical link path that is to be started. The MSNAME keyword can be generic.

NODE

Allows IMS to accept logons from VTAM-attached terminals. It only operates if the node is disconnected, idle, and stopped. Any terminals in MFSTEST mode, exclusive mode, or signon status are reset. The user is not signed off and the conversations are terminated. /START NODE is valid for temporary nodes, even though the node is not logged on.

The /START NODE command is only effective for existing nodes.

If an ETO logon terminal is stopped, use the /OPNDST NODE USER command to start it, not the /START NODE command.

If Fast Path is active for a specified physical terminal or node, /START LINE and /START NODE cannot reset terminal response mode. The /DEQUEUE command must be entered to discard Fast Path output before using /START.

The /START NODE nodename USER username command applies to ISC sessions only, and it is used to start a half-session allocated to USER username for NODE nodename. The USER keyword when used with the NODE keyword affects the specified half-session. When the USER keyword is omitted, all half-sessions of the specified node are affected.

The NODE parameter can be generic if the USER keyword is not present. The generic parameter specifies nodes that already exist.

/START NODE no longer resets response mode, test mode, and preset mode, since these statuses are no longer significant and therefore no longer carried across logon or restart. MFSTEST mode (at the node level) and exclusive mode are still reset.

If global resource information is kept in Resource Manager, the /START NODE command allows a node to logon to any IMS in the IMSplex and resets MFSTEST mode and exclusive mode. If the node no longer has significant status, it is deleted from Resource Manager.

OLDS

Indicates that either a previously stopped OLDS is to be started or that IMS is to add a new OLDS log data set. If a new OLDS is being added, olds# is an OLDS identifier that is defined by the DFSMDA macro specification. If in dual mode, both primary and secondary OLDSs are started. olds# must be 00-99.

When using /START OLDS, an OLDS must be defined in the DFSMDA macro, even if it is allocated in JCL.

OTMA

Causes IMS to join the XCF group for the IMS Open Transaction Manager Access (OTMA).

/START OTMA command processing is as follows:

1. IMS joins the XCF group.
2. Following a successful Client-Bid, IMS sends an ACK message to the OTMA client.
3. IMS begins sending all Commit-then-Send (commit mode 0) output messages to the OTMA client.

PROGRAM

Specifies the application program that is to be started. This command also clears the indicator preventing a program from scheduling when I/O prevention has not completed. The integrity of a GSAM database residing on DASD can be affected if I/O prevention has not been done on a failing active system.

/START PROGRAM does not start a CPI Communications driven transaction program.

REGION

Specifies the set of message processing region JCL to be passed to z/OS. If no member name is specified, the default member name is used.

IMS dependent regions of the same type (MPP, BMP, or IFP) can share a PROCLIB member containing the startup JCL for the type of region. Use the JOBNAME or LOCAL keywords of the /START REGION command to allow IMS to set (or override) the IMS ID for the dependent region to match the IMS ID of the IMS that processes the command.

Restriction: The JCL for the region to be started must include the IMSID= execution parameter.

Use the JOBNAME keyword to override the job name on the JOB statement of the default or specified JCL member for a dependent region.

If you specify the LOCAL keyword, IMS overrides the symbolic IMSID parameter in the JCL of the default or specified member. LOCAL is the default if you specify the JOBNAME keyword.

When the LOCAL or JOBNAME keywords are specified on the /START REGION command, the PROCLIB member must be a job that runs a procedure to start the dependent region. The procedure cannot be an instream procedure. For example, suppose that the /START REGION command is entered in one of the following formats:

```
/START REGION member_name LOCAL
/START REGION member_name JOBNAME job_name
/START REGION member_name JOBNAME job_name LOCAL
```

In these instances, *member_name* is a job that runs a procedure to start the dependent region and has the following format:

```
//job_name JOB ... (parameters)...
//      EXEC proc_name,
//       IMSID=xxxx
```

The operator can start more dependent regions than were specified in the IMS system definition or the EXEC parameter, up to 999. A request to start more regions than the system-definition value (but less than or equal to 999) might be rejected if resources are not available.

The /START REGION command is not mirrored on the XRF alternate subsystem. You must enter this command on the alternate subsystem if you want it to affect the alternate subsystem.

RTCODE
Specifies the Fast Path routing codes to be activated and allows transactions associated with the routing codes to be processed.

SB
Dynamically allows sequential buffering. This command does not affect sequential buffering applications scheduled before this command was issued.

SERVGRP
Starts communications between the entering service group and the service group at the other site in an RSR complex. If the subsystem is not currently identified to the transport manager, an attempt to identify precedes an attempt to connect to the other subsystem. /START SERVGRP is supported on an active IMS subsystem and an RSR tracking subsystem.

The /START SERVGRP command is not normally needed for an active subsystem, since the logger normally attempts to identify to transport manager at each OLDS switch to establish connections with the other subsystem. However, the operator may want to trigger this process between OLDS switches; for instance if a network outage between the active and tracking sites has been repaired and the operator does not want to wait until the next OLDS switch to re-establish communications.

Successful completion of the syntax checking of the /START SERVGRP command results in the DFS058 START COMMAND COMPLETED message, although processing of the command continues asynchronously.

SLDSREAD
Indicates whether IMS is enabled to retrieve records from both a system log data set (SLDS) and OLDS or OLDS only. The default is that SLDSREAD is enabled.

SSM
Allows external subsystem connection processing to occur even though the option was not requested when IMS was started.

The syntax and usage of the SSM keyword are the same as the SSM=EXEC parameter that can be specified on the IMS startup JCL. The SSM keyword is followed by a 1 to 4 character identifier. IMS concatenates the SSM identifier to the IMSID to create an SSM PROCLIB member name. The SSM Proclib member is then used for external subsystem processing.

The SSM keyword is not valid if either of the following conditions apply:

- The SSM= keyword is specified in the EXEC parameters of the startup JCL.
- The /START SUBSYS SSM command has been previously issued.

SUBSYS
Specifies the external subsystem to which IMS is to connect. This command can also be used to dynamically reconfigure existing subsystem definitions. The installation can start IMS with the subsystem PROCLIB member defining one subsystem. The PROCLIB member can then be changed or added to. The operator can then /STOP the existing subsystem connections or only the one that has changed. By issuing the /START SUBSYS command, IMS will pick up the new or changed definitions and attempt to connect to those subsystems.

The /START SUBSYS ALL command connects IMS to all external subsystems. Also, the SSM keyword can be used with the /START SUBSYS command.

If the subsystem connection was abnormally terminated, IMS puts the connection in a stopped state. In this instance, the /START command must be used to reestablish the connection.

SURVEILLANCE
Is used in an XRF environment to start the operation of the IMS surveillance function. When surveillance is on for a function, potential failures of the active system are detected. Based on information from surveillance, the alternate system either requests a takeover or informs the operator of the potential failure. The following are the surveillance functions to be started:

LNK	ISC link
RDS	Restart data set
LOG	System log
ALL	Same as specifying LNK, RDS, and LOG

The surveillance function is generally started during IMS system definition by using the /START SURVEILLANCE control statement. ALL is the default.

THREAD
Specifies the set of message processing region JCL to be passed to z/OS. If no member name is specified, the default member name is used.

The /START THREAD command is used only for DEDB utility regions. BMP regions are started by JCL. CCTL threads are started automatically at connection: first to the MINTHREAD value, and later (on demand) to MAXTHREAD value.

TMEMBER
Causes IMS to send an Open Transaction Manager Access (OTMA) command to OTMA clients to request that input resume for the specified transaction pipe name. IMS then resumes sending output to the OTMA client.

TPIPE
When it is used with the /START command, TPIPE causes IMS to resume sending output to OTMA clients.

Restriction: If a transaction pipe has a resynchronization pending status, IMS does not start the transaction pipe.

TRANSACTION
Specifies the transactions to be started.

The /START TRANSACTION ALL CLASS cls# command causes all transactions associated with the specified class to be started.

If a transaction that has messages on the suspend queue is started, the suspend queue associated with the transaction will automatically be transferred to the normal queue.

The TRANSACTION keyword can be generic where the generic parameter specifies transactions that already exist.

TRKAUTOARCH
Indicates that the RSR tracking subsystem is to initiate a request to start the automatic archiving of the tracking log data sets. This keyword allows the user to start automatic archive after it has been terminated following archive data set full conditions.

Successful completion of the syntax checking of the /START TRKAUTOARCH command results in the DFS058I START COMMAND COMPLETED message, although processing of the command continues asynchronously.

USER
Without the NODE keyword, USER specifies the ISC user or the dynamic user to start. The USER parameter can be generic where the generic parameter specifies users that already exist.

/START USER applies only to users that are stopped and signed off. The /START USER command also terminates any active conversations before starting the user.

/START USER removes the user from MFSTEST mode and exclusive mode. If the USER structure is temporary and the status conditions that caused the creation of the structure have been reset, the temporary user is deleted at the next simple checkpoint.

For ISC users, the user is started and made available for allocation.

For dynamic users, the user is started and made available for signon.

/START USER no longer removes the user from response mode, test mode, and preset mode. MFSTEST mode can now be associated with the node and with the user. MFSTEST mode (at the user level) and exclusive mode are still reset. The other statuses are no longer significant and therefore not carried across signon or restart.

If global resource information is not kept in Resource Manager, the /START USER command allows a user to signon to the local IMS. If global resource information is kept in Resource Manager, the /START USER command allows a user to signon to any IMS in the IMSplex.

VGRS
Causes the IMS subsystem to join a VTAM generic resource group. The command is rejected if the VTAM ACB is closed (usually the result of a /STOP DC command).

The GRSNAME keyword allows you to specify the generic resource name if the IMS subsystem does not have one. The IMS subsystem already has a generic resource name if it has the GRSNAME= keyword specified on its EXEC statement.

WADS
Indicates that either a previously stopped WADS is to be started or that IMS is to add a new WADS to the pool of available WADSs. If a new WADS is being added, wads# is a WADS identifier that is defined by the DFSMDA macro specification. wads# must be 0-9.

When using /START WADS, a WADS must be defined in the DFSMDA macro, even if it is allocated in JCL.

XRCTRCK
Results in calls to the log router to initiate or terminate XRC tracking. It is only valid on a tracking IMS system.

Examples

Example 1 for /START Command

Entry ET:

```
/START AREA DB1AREA0 DB1AREA1
```

Response ET:

```
DFS058I  START COMMAND IN PROGRESS
DFS0488I  START COMMAND COMPLETED.  AREA=DB1AREA0
DFS0488I  START COMMAND COMPLETED.  AREA=DB1AREA1
```

Explanation: DEDB areas DB1AREA0 and DB1AREA1 are started.

Example 2 for /START Command

Entry ET:

```
/START AUTOARCH 4
```

Response ET:

```
DFS058I  START COMMAND COMPLETED
```

Explanation: Automatic archiving will be initiated after 4 OLDS data sets are filled.

Example 3 for /START Command

TSO SPOC input:

```
STA DB BANKATMS BANKTERM BANKLDGR BE3ORDER
```

TSO SPOC output:

```
SYS3     DFS0488I  STA COMMAND COMPLETED. DBN= BANKATMS RC=04
    SYS3     DFS0488I  STA COMMAND COMPLETED. DBN= BANKTERM RC=04
    SYS3     DFS0488I  STA COMMAND COMPLETED. DBN= BANKLDGR RC=04
    SYS3     DFS0488I  STA COMMAND COMPLETED. DBN= BE3ORDER RC=08
    IMS3     DFS0488I  STA COMMAND COMPLETED. DBN= BANKATMS RC=04
```

```
    IMS3     DFS0488I  STA COMMAND COMPLETED. DBN= BANKTERM RC=04
    IMS3     DFS0488I  STA COMMAND COMPLETED. DBN= BANKLDGR RC=04
    IMS3     DFS0488I  STA COMMAND COMPLETED. DBN= BE3ORDER RC=08
```

OM API input:

```
CMD (STA DB BANKATMS BANKTERM BANKLDGR BE3ORDER )
```

OM API output:

```
<?xml version="1.0"?>
<!DOCTYPE imsout SYSTEM "imsout.dtd">
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2002.197 21:59:29.210362</statime>
<stotime>2002.197 21:59:30.213238</stotime>
<staseq>B7EFC01B367FAE02</staseq>
<stoseq>B7EFC01C2B576D8F</stoseq>
<rqsttkn1>USRT005 10145929</rqsttkn1>
<rc>0200000C</rc>
<rsn>00003008</rsn>
</ctl>
<cmderr>
<mbr name="SYS3    ">
<typ>IMS     </typ>
<styp>DBDC    </styp>
<rc>00000014</rc>
<rsn>00005050</rsn>
</mbr>
<mbr name="IMS3    ">
<typ>IMS     </typ>
<styp>DBDC    </styp>
<rc>00000014</rc>
<rsn>00005050</rsn>
</mbr>
</cmderr>
<cmd>
<master>SYS3    </master>
<userid>USRT005 </userid>
<verb>STA </verb>
<kwd>DB              </kwd>
<input>/STA DB BANKATMS BANKTERM BANKLDGR BE3ORDER   </input>
</cmd>
<msgdata>
<mbr name="SYS3    ">
<msg>DFS0488I  STA COMMAND COMPLETED. DBN= BANKATMS RC=04</msg>
<msg>DFS0488I  STA COMMAND COMPLETED. DBN= BANKTERM RC=04</msg>
<msg>DFS0488I  STA COMMAND COMPLETED. DBN= BANKLDGR RC=04</msg>
<msg>DFS0488I  STA COMMAND COMPLETED. DBN= BE3ORDER RC=08</msg>
</mbr>
<mbr name="IMS3    ">
<msg>DFS0488I  STA COMMAND COMPLETED. DBN= BANKATMS RC=04</msg>
<msg>DFS0488I  STA COMMAND COMPLETED. DBN= BANKTERM RC=04</msg>
<msg>DFS0488I  STA COMMAND COMPLETED. DBN= BANKLDGR RC=04</msg>
<msg>DFS0488I  STA COMMAND COMPLETED. DBN= BE3ORDER RC=08</msg>
</mbr>
</msgdata>
</imsout>
```

Explanation: The START DB command is routed from OM to the two active IMSs - SYS3 and IMS3. The response from both IMSs is returned to OM. The databases BANKATMS, BANKTERM, BANKLDGR, and BE3ORDER are started at both IMSs.

Example 4 for /START Command

Entry ET:

```
/START CLASS ALL
```

Response ET:

```
DFS058I  START COMMAND COMPLETED
```

Explanation: All classes of transactions are made available for scheduling into message processing regions.

Example 5 for /START Command

Entry ET:

```
/START DATABASE TREEFARM
```

Response ET:

```
DFS058I  (time stamp) START COMMAND IN PROGRESS
DFS0488I START COMMAND COMPLETED. DBN=TREEFARM RC=0.
```

Explanation: Database TREEFARM is started.

Example 6 for /START Command

Entry ET:

```
/START LINE 4 PTERM 1, 2
```

Response ET:

```
DFS058I  START COMMAND COMPLETED
```

Response RT:

```
DFS059I  TERMINAL STARTED
```

Explanation: Physical terminals 1 and 2 on line 4 are started.

Example 7 for /START Command

Entry ET:

```
/START LINE 4,5,6,7,8,9,10,11
```

Response ET:

```
DFS058I  START COMMAND COMPLETED
```

Response RT:

```
DFS059I  TERMINAL STARTED
```

Explanation: Lines 4,5,6,7,8,9,10, and 11 are started.

Example 8 for /START Command

Entry ET:

```
/START LINE 4 5 6 700
```

Response ET:

```
DFS058I  START COMMAND COMPLETED EXCEPT LINE 5 700
```

Response RT:

```
DFS059I  TERMINAL STARTED
```

Explanation: Lines 4 and 6 are started. The /DISPLAY LINE command can be used to determine why line 5 did not start successfully. (700 is an invalid line number.)

Example 9 for /START Command

Entry ET:

```
/START LTERM APPLE, TREE, FRUIT
```

Response ET:

```
DFS058I  START COMMAND COMPLETED
```

Response RT:

```
DFS059I  TERMINAL STARTED
```

Explanation: Logical terminals APPLE, TREE, and FRUIT are started.

Example 10 for /START Command

Entry ET:

```
/START MSNAME CHICAGO
```

Response ET:

```
DFS058I  START COMMAND COMPLETED
```

Explanation: A logical link path associated with the name CHICAGO is started.

Example 11 for /START Command

Entry ET:

```
/START NODE HARRY
```

Response ET:

```
DFS058I  START COMMAND COMPLETED
```

Explanation: The physical terminal associated with the node HARRY is started.

Example 12 for /START Command

Entry ET:

```
/STA OTMA
```

Response ET:

```
DFS2360I 14:02:53 XCF GROUP JOINED SUCCESSFULLY. SYS3
DFS058I 14:02:53 START COMMAND COMPLETED   SYS3
DFS996I *IMS READY*  SYS3
```

Example 13 for /START Command

Entry ET:

```
/START OLDS 09
```

Response ET:

```
DFS058I  START COMMAND IN PROGRESS
```

Explanation: OLDS data set DFSOLP09 (DFSOLS09) will be started for logging.

Example 14 for /START Command

Entry ET:

```
/START PROGRAM ALL
```

Response ET:

```
DFS058I  START COMMAND COMPLETED
```

Explanation: All application programs are started.

Example 15 for /START Command

Entry ET:

```
/START PROGRAM APPLETRE
```

Response ET:

```
DFS058I  START COMMAND COMPLETED
```

Explanation: Application program APPLETRE is started.

Example 16 for /START Command

Entry ET:

```
/START REGION
```

Response ET:

```
DFS058I  START COMMAND IN PROGRESS
```

Response ET:

```
DFS551I IFP|MESSAGE|BATCH  REGION XXXXXXXX STARTED. ID=yy TIME=zzzz
CLASSES=xxx,xxx,xxx,xxx
```

Explanation: One message region or batch region (ID=yy) is started at TIME=zzzz. The transactions associated with the classes listed in the response can now be scheduled.

Example 17 for /START Command

Entry ET:

```
/START REGION IMSWT000
```

Response ET:

```
DFS058I  START COMMAND IN PROGRESS
```

Explanation: The JCL stored as member IMSWT000 is used to start the spool SYSOUT utility for the data sets associated with the spool line corresponding to the IMSWT000 procedure.

Example 18 for /START Command

Entry ET:

```
/START REGION MEMABC
```

Response ET:

```
DFS058I  START COMMAND IN PROGRESS
```

Response ET:

```
DFS551I IFP|MESSAGE|BATCH  REGION XXXXXX STARTED. ID=yy TIME=zzzz
CLASSES=xxx,xxx,xxx,xxx
```

Explanation: The JCL stored as member XXXXXX is used to start a message processing region or batch message processing region with the classes specified by the EXEC statement parameters in MEMABC.

Example 19 for /START Command

Entry ET:

```
/START RTCODE ALL
```

Response ET:

```
DFS058I  START COMMAND COMPLETED
```

Explanation: All the Fast Path routing codes are activated. Transactions associated with these routing codes can now be processed.

Example 20 for /START Command

Entry ET:

```
/START SB
```

Response ET:

```
DFS058I  START COMMAND COMPLETED
```

Entry ET:

```
/DISPLAY POOL DBAS
```

Response ET:

```
SEQUENTIAL BUFFERING:  STATUS = NOT-STOPPED
  MAX      N.A.  FREE   N.A.   CURR   160K  HIGH  320K
DATABASE BUFFER POOL: SIZE   67584
  REQ1       0 REQ2     0 READ     0 BISAM     0 WRITES     0
  KEYC       0 LCYL     0 PURG     0 OWNRR     0 ERRORS 00/00
  DATABASE BUFFER POOL:  BSIZE  12288
    RRBA     0 RKEY     0  BFALT    0 NREC  0 SYN PTS     0
    NMBUFS  29 VRDS     0   FOUND   0 VWTS  0  ERRORS 00/00
  DATABASE BUFFER POOL:  BSIZE  356352
    RRBA     0 RKEY     0  BFALT    0 NREC  0 SYN PTS     0
    NMBUFS  29 VRDS     0  FOUND    0 VWTS  0  ERRORS 00/00
  *86253/104547*
```

Explanation: Sequential buffering is started.

Example 21 for /START Command

Entry ET:

```
/START SUBSYS ABC
```

Response ET:

```
DFS058I  START COMMAND COMPLETED
```

Explanation: IMS has established a connection to the requested subsystem. It is likely that an external subsystem (not CCTL) connection message will be received at this time. If this is not the case, the /DISPLAY command can be used.

Example 22 for /START Command

Entry ET:

```
/sta tmember client1 tpipe tpipesy
```

Response ET:

```
DFS058I 15:39:40 START COMMAND COMPLETED   SYS3
DFS996I *IMS READY*  SYS3
```

Example 23 for /START Command

Entry ET:

```
/START TRANSACTION ALL CLASS 6
```

Response ET:

```
DFS058I  START COMMAND COMPLETED
```

Explanation: All transactions associated with class 6 are started.

Example 24 for /START Command

Entry ET:

```
/START TRANSACTION PIT, SEED
```

Response ET:

```
DFS058I  START COMMAND COMPLETED
```

Explanation: Transactions PIT and SEED are started.

Example 25 for /START Command

Entry ET:

```
/DISPLAY USER IMSUS01 IMSUS02
```

Response ET:

```
USER     ENQCT DEQCT   QCT
IMSUS01      0     0     0 STOPPED
IMSUS02      0     0     0 STOPPED
*91091/111727*
```

Entry ET:

```
R 38,/START USER IMSUS01
```

Response ET:

```
DFS058I 11:19:05 START COMMAND COMPLETED
```

Entry ET:

```
/DISPLAY USER IMSUS01 IMSUS02
```

Response ET:

```
USER     ENQCT DEQCT   QCT
IMSUS01      0     0     0
IMSUS02      0     0     0 STOPPED
*91091/113038*
```

Entry ET:

```
/START USER APPLE*
```

Response ET:

```
DFS3633 11:19:35 GENERIC PARAMETER RESOURCES NOT FOUND, NO ACTION TAKEN
```

Example 26 for /START Command

Entry ET (at the active site):

```
/START SERVGRP
```

Response ET (to the active subsystem):

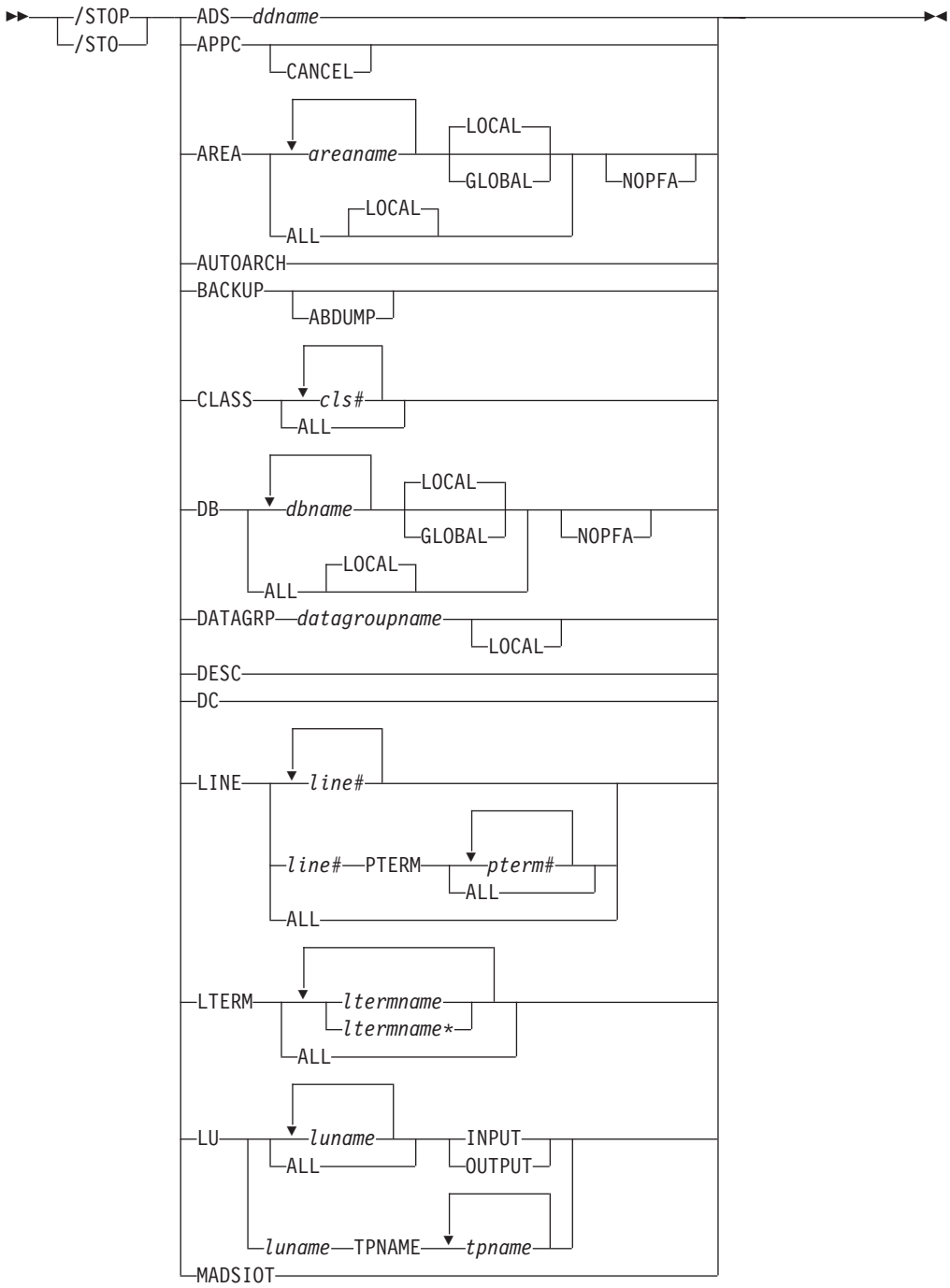
```
DFS058 START COMMAND COMPLETED
```

Explanation: Communications between the subsystem at the active site and the subsystem at the RSR tracking site are started.

Chapter 59. /STOP

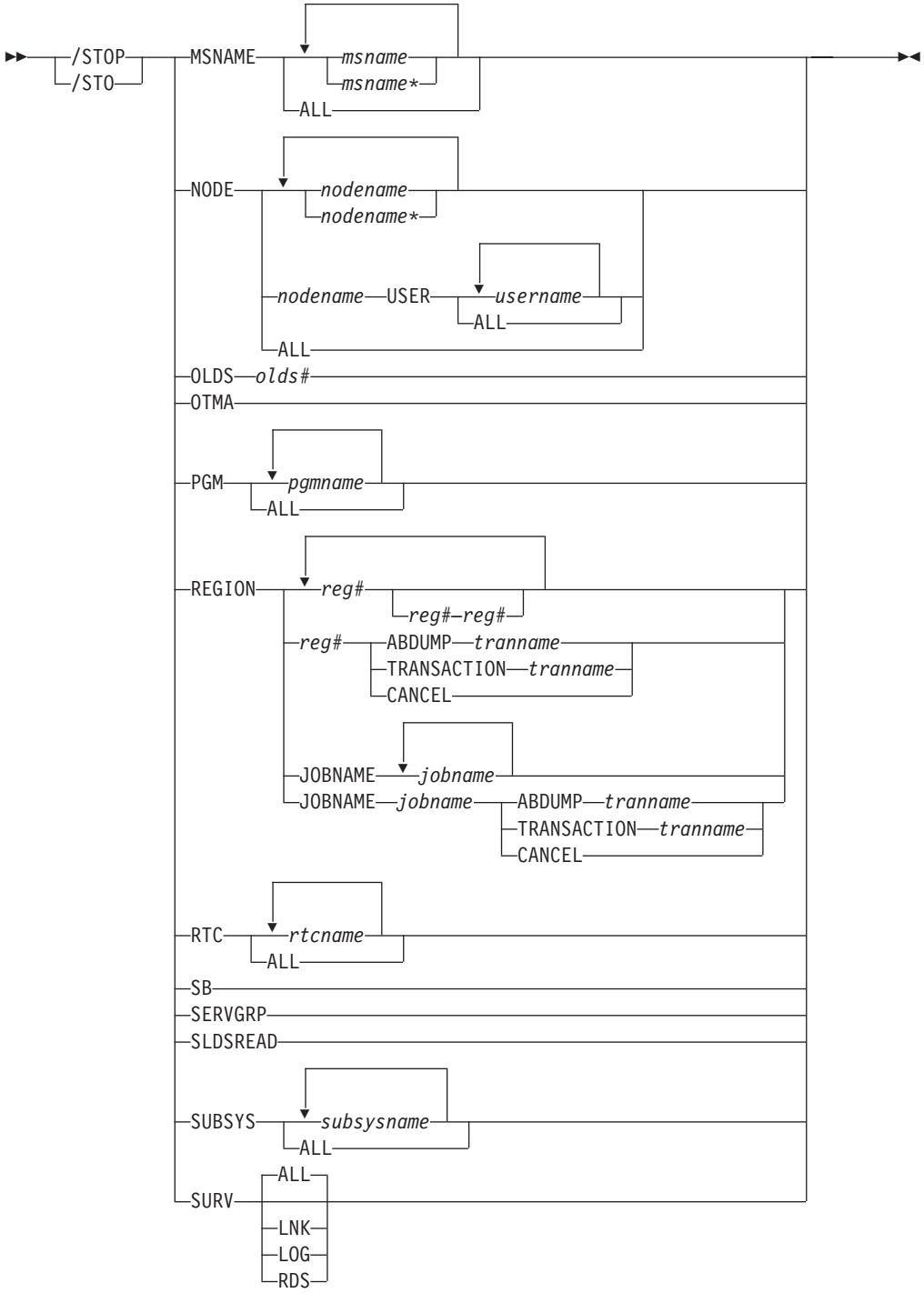
Format

/STOP Command: ADS through MADSIOT

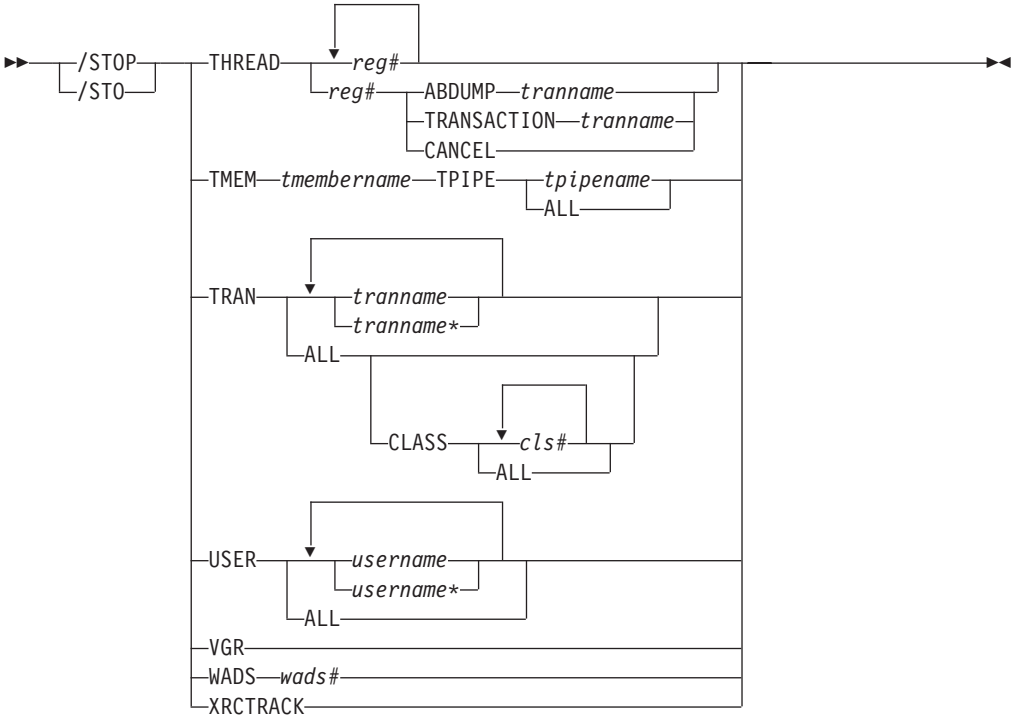


STOP Command: MSNAME through SURV

© Copyright IBM Corp. 1974, 2003



STOP Command: THREAD through XRCTRACK



Environments and Keywords

Table 150 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 150. Valid Environments for the /STOP Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/STOP	X	X	X
ABDUMP	X	X	X
ADS	X	X	
APPC	X		X
AREA	X	X	
AUTOARCH	X	X	X
BACKUP	X		X
CANCEL	X	X	X
CLASS	X		X
DB	X	X	
DATAGRP	X	X	
DESC	X		X
DC	X		X
GLOBAL	X	X	
INPUT	X		X
JOBNAME	X	X	X
LINE	X		X
LOCAL	X	X	

Table 150. Valid Environments for the /STOP Command and Keywords (continued)

Command / Keywords	DB/DC	DBCTL	DCCTL
LTERM	X		X
LU	X		X
MADSIOT	X	X	
MSNAME	X		X
NOPFA	X	X	
NODE	X		X
OLDS	X	X	X
OTMA	X		X
OUTPUT	X		X
PGM	X	X	X
PTERM	X		X
REGION	X	X	X
RTC	X		X
SB	X	X	
SERVGRP	X	X	X
SLDSREAD	X	X	X
SUBSYS	X	X	X
SURV	X		X
THREAD	X		X
TMEM	X		X
TPIPE	X		X
TPNAME	X		X
TRAN	X		X
USER	X		X
VGR	X		X
WADS	X	X	X
XRCTRACK	X	X	X

Usage

The /STOP command stops:

- The sending, receiving, or queuing of output messages to a particular communication line, terminal, user, or logical path
- The scheduling or queuing of messages containing a specific transaction code
- The execution of a specific program
- The use of a given database

For VTAM nodes, the currently connected terminal is disconnected. All further logons are rejected until the node is the subject of a /START or /RSTART command.

The /STOP command can be used on HALDBs. For more information see Appendix H, “High Availability Large Database Commands,” on page 867

/STOP validity checks all parameters entered by the terminal operator. If an error is detected on parameters that are independent of one another, only the invalid parameters are indicated as being in error and the /STOP command processes the rest of the parameters.

/STOP can be used to reset conditions previously established by the /START, /RSTART, /PSTOP, /PURGE, or /MONITOR commands.

ADS

Specifies the area data set to be closed and deallocated. The AREA is not stopped as long as at least one data set in the AREA remains open. /STOP ADS is rejected if the specified ADS is the last data set available in the AREA.

Although the /STOP ADS command has no option of LOCAL/GLOBAL, if the DEDB area is shared at the block level, the response is the same as if GLOBAL were specified.

The output of the /STOP ADS command is changed when the command is entered through the OM API. In this case, the DFS058I message is not returned to OM. The command response returned to OM contains one or more of the following messages as appropriate.

Fast Path messages: DFS140I, DFS0488I, DFS0666I, DFS1407I, DFS3720I, DFS3721I, DFS3771I

APPC

Instructs IMS to stop scheduling transactions from LU 6.2 devices. /STOP APPC can be used in a transient stopped state. It causes remote LU 6.2 devices to receive a sense code of TP_Not_Available_No_Retry. This is likely to lead to further attempts to access IMS.

/STOP APPC sets the desired status to STOPPED. The current status is set to STOPPED or FAILED according to the response from APPC/MVS.

CANCEL

Causes APPC/MVS to initiate a shutdown request when a long stopped period is anticipated, for example, at the end of the day's processing. All remote LU 6.2 devices receive a sense code of TP_Not_Available_No_Retry. The remote LU 6.2 devices stop trying to access this application.

/STOP APPC CANCEL sets the desired status to CANCEL according to responses from APPC/MVS. If the desired status is DISABLED. then IMS rejects /STOP APPC CANCEL[2] when it is entered.

AREA

Specifies that the data sets associated with this area are closed. Use the /DISPLAY AREA command to determine if the area is stopped or closed. If the area is stopped, the area must be made available using the /START AREA command. In z/OS, all the data sets are deallocated. If the system processes a /STOP AREA command during HSSP processing, the area will be released after the current commit processing completes. Any image copy option in effect at /STOP time can affect the continued system operation. All virtual storage option (VSO) DEDB areas that are being stopped and that are in a z/OS data space are removed from the data space and updates are written out to DASD.

2. The sense code returned to the LU 6.2 remote device for an incoming ATTACH to a stopped APPC/IMS system is determined by APPC/MVS, and it might differ from release to release. In general, the remote LU 6.2 application should wait for a period of time after rejection before any attempts to reestablish a session with IMS.

The output of the /STOP AREA command is changed when the command is entered through the OM API. In this case, the DFS058I message is not returned to OM. For commands that specify GLOBAL, only the command master returns the asynchronous messages to OM. When a command is processed with the LOCAL keyword, all IMSs are able to return the asynchronous messages to OM. The command response returned to OM contains one or more of the following messages as appropriate.

Fast Path messages: DFS140I, DFS170I, DFS0488I, DFS0666I, DFS1407I, DFS3062I, DFS3342I, DFS3720I, DFS3824I

/STOP AREA is not supported on an RSR tracking subsystem.

GLOBAL
Specifies when an IRLM is active and that the command applies to all subsystems sharing the database or area.

The GLOBAL keyword and the ALL parameter are mutually exclusive. If both keywords are specified, the command is rejected. The GLOBAL keyword requires that IRLM be active. If IRLM is not active, the command is rejected. DBRC is informed that the database or area has been stopped and will update the RECON data set to indicate the stopped condition.

The GLOBAL keyword is not supported on an RSR tracking subsystem.

If the GLOBAL keyword on a command is entered from an OM API, the command should only be routed to one IMS system in the IMSplex. The IMS that receives the command from OM will make DBRC calls to update the RECON with GLOBAL status. It will also request IRLM NOTIFY to route and process the command on sharing IMS systems, and then process the command locally.

Messages produced on the NOTIFIED systems will appear only on the system console and will not be routed back to the OM API which originally entered the command.

If multiple IMS systems have been explicitly specified in the route list, the master IMS system will process the command as described above. However, the non-master IMS systems, to which OM routes the command, will reject the command with the following return and reason code listed in Table 151:

Table 151. Return and reason code for GLOBAL keyword issued from the OM API

Return Code	Reason Code	Meaning
X'00000004'	X'00001000'	The command contained the GLOBAL keyword and was routed to more than one IMS system in the IMSPLEX. The non-master IMS systems will reject this command when OM routes the command to them. The master IMS system will process this command and use IRLM NOTIFY to route and process the command on the non-master IMS systems. See the discussion under the GLOBAL keyword.

LOCAL
Specifies that the command only applies to the subsystem in which the command is entered. This command does not affect any other subsystem sharing the database or area. LOCAL is the default.

NOPFA
Specifies that DBRC is not notified that the database or area has changed status. You can use this keyword when you need to authorize the database for use after it is offline, for example, for offline utilities. By using this keyword, DBRC does not prevent further authorizations for the database or area.

Recommendation: Before restarting the database or area, issue this command **without** the NOPFA keyword to inform DBRC of the change in status for the database or area.

AUTOARCH
Specifies that automatic archiving is to be stopped.

BACKUP
Terminates the alternate system in an XRF environment. This command must be entered on the alternate system. The ABDUMP keyword results in a dump of the alternate system.

CLASS
Prevents further scheduling of application programs for the designated class.

All regions currently handling transactions assigned to the specific class are allowed to run until the limit count is reached (MPPs) or the input queue contains no more messages (BMPs and MPPs).

The region is not allowed to wait for the next message (wait-for-input mode). Instead a QC status code (no more messages) is returned to the application (MPPs).

If the region is already scheduled and waiting for the next message (wait-for-input mode) when the command is entered, the region is notified and a QC status code is returned to the application. (MPPs).

A batch message processing region (BMP) scheduled against wait-for-input (WFI) transactions returns a QC status code (no more messages) for /PSTOP REGION, /DBD, /DBR, or /STA commands only.

DATABASE
Prevents subsequently scheduled programs from accessing the database, without affecting currently scheduled programs or closing the database.

If the database is a DEDB or MSDB, programs using the database will not be scheduled. For other databases, the programs will still be scheduled but a call against the database will result in either a 3303 pseudoabend, or a BA status code, if the INIT call was issued.

If the database is stopped after the region is scheduled, the region is not allowed to wait for the next message (wait-for-input mode). If there are no more messages available for the region to process, a QC status (no more messages) will be returned to the application (MPPs). If the region is already scheduled and waiting for the next message (wait-for-input mode) when the command is entered, the region is notified and a QC status code is returned to the application (MPPs).

A batch message processing region (BMP) scheduled against wait-for-input (WFI) transactions returns a QC status code (no more messages) for /PSTOP REGION, /DBD, /DBR, or /STA commands only.

In an IFP region, the /STOP command has no effect until the region is started again.

See the AREA keyword for a description of the LOCAL, GLOBAL, and NOPFA keywords.

For DBCTL, when CCTL schedules a PSB, the DBCTL thread SCHED request defines the thread as LONG or SHORT. If the database is currently scheduled to a LONG thread, the command is rejected; otherwise, the thread is allowed to complete before the database is acted upon. This results in either a commit point or transaction termination.

For the results of issuing this command on a shared secondary index, see Appendix D, “Shared Secondary Index Database Commands,” on page 849

/STOP DATABASE is not supported on an RSR tracking subsystem.

The output of the/STOP DATABASE command is changed when the command is entered through the OM API. In this case, the DFS058I message is not returned to OM. The command response returned to OM contains one or more of the following messages as appropriate to the database type and the command completion.

Full Function Database messages: DFS132, DFS160, DFS216, DFS0488I, DFS1407, DFS2026, DFS3318I, DFS3466I

Fast Path Database messages: No unique messages are returned.

DATAGROUP

Specifies groups of DL/I databases, Fast Path DEDBs, and Fast Path areas to be stopped. Data groups are logical groupings of databases and areas; they allow simplified command processing for databases and areas. You define a database group in the RECON data set using the INIT.DBDSGRP command with parameters GRPNAME and DBGRP. DATAGROUP is not valid on RSR tracking subsystems.

Related Reading: See “Group Parameters” on page 20 for more information on defining data groups.

During /STOP DATAGROUP processing, all virtual storage option (VSO) DEDBs that are in a z/OS data space are removed from the data space and updates are written out to DASD.

Recommendation: Although you can use DBDS groups as well as database groups for this command, you should use database groups whenever possible to eliminate the overhead of converting the DBDS group to a database group.

DESCRIPTOR

Defines the LU62 descriptors from DFS62DTx PROCLIB member to IMS.

DC

Prohibits you from logging on to VTAM and ensures that all VTAM node sessions have terminated before IMS issues the DFS2111I message which says the ACB is closed.

Note: There are two ACBs if MNPS for XRF is used. If you use XRF with MNPS, both the APPLID and MNPS ACB are closed. If you are not using XRF or using XRF without MNPS, there is only one ACB, the VTAM ACB, which is closed. APPLID ACB is the same as VTAM ACB.

The /STOP DC command can be used either before or after the /CLSDST NODE or /STOP NODE command, the only difference being that logons can still occur if the /STOP DC command is not entered. However, the command cannot start or complete processing if the VTAM ACBs (APPLID and MNPS) are not open or the VTAM nodes remain active. If the nodes are active, the /CLSDST NODE or

2 /STOP NODE command must be issued to close the nodes; in some cases, a
2 /IDLE NODE command can be issued to cause an OS VTAM VARY command to
2 be issued against any nodes that remain connected.

LINE

Stops message queuing for lines and stops the sending and receiving of messages over the lines. However, lines are not considered stopped unless they are stopped and idle. Use /DISPLAY LINE to verify line status.

/STOP LINE PTERM ensures that no input messages from any of the specified terminals assigned to the specified lines will be received by IMS after the command is issued.

/STOP LINE resets preset mode, test mode, response mode, lock pterm, lock lterm, pstop lterm, and purge lterm because these statuses are not significant and therefore are not kept after a /START LINE or restart.

LTERM

Specifies the LTERM that is to be stopped. The /STOP LTERM command with a logical terminal that is in a QLOCKED state does not reset the QLOCK state, but puts the LTERM in a STOPPED and QLOCKED state.

If IMS internally resets the QLOCK condition, the LTERM remains in a STOPPED state. (QLOCK indicates that the LTERM is locked from sending any further output or from receiving input that can create additional output for the same LTERM until the state is reset by a specific request received on the session.)

/STOP LTERM is rejected for remote logical terminals.

The LTERM parameter can be generic where the generic parameter specifies LTERMs that already exist.

If global resource information is kept in Resource Manager, the /STOP LTERM command stops messages from being queued to the lterm anywhere in the IMSplex and the change is reflected both in Resource Manager and in the local IMS system.

LUNAME TPNAME

Specifies a particular LU name that is to be stopped.

Specifying the keyword INPUT with the LUNAME TPNAME keyword stops an luname for any input and synchronous outbound activities. Specifying the parameter ALL with INPUT causes all future LU 6.2 input and synchronous outbound activities to be stopped as well.

Specifying the keyword OUTPUT with the LUNAME TPNAME keyword stops an luname for any asynchronous outbound activities. Specifying the parameter ALL with OUTPUT causes all future LU 6.2 asynchronous outbound activities to be stopped as well.

Specifying neither INPUT nor OUTPUT is the same as specifying both INPUT and OUTPUT. The LU name is stopped for any input, and both synchronous and asynchronous outbound activities. Specifying the parameter ALL in this case stops all future LU 6.2 inbound activities, synchronous and asynchronous outbound activities.

A network-qualified LU name is optional for the LUNAME keyword. If the LU name is not network-qualified and no TP name is specified, all network-qualified LU names whose LU names match the LU name specified are also stopped.

/STOP LUNAME TPNAME stops a particular TP name of the LU name specified.

If the specified resource does not exist, a structure is created to retain the status.

MADSIOT

Allows users to disable the MADS I/O timing function in a MADS I/O timing enabled environment. When /STOP MADSIOT completes normally, the following message is returned to the operator's console:

```
DFS12761 MADS I/O TIMING FUNCTION STOPPED SUCCESSFULLY
```

/START MADSIOT allows users to resume the MADS I/O timing function.

If MADS I/O Timing list structure is not defined in DFSVSMxx, the command will be rejected. If MADS I/O Timing function is already disabled, the command will be ignored. If MADS I/O Timing function is enabled and all sharing partners successfully disconnect from MADS I/O Timing list structure on the coupling facility, the command will complete successfully; if any sharing partners fails to disconnect to MASD I/O Timing list structure, the command will fail.

The output of the/STOP MADSIOT command is changed when the command is entered through the OM API. In this case, the DFS058I message is not returned to OM. The command response returned to OM contains one or more of the following messages as appropriate.

Fast Path messages: DFS0023I, DFS0008I, DFS1271I, DFS1276I, DFS1275E, DFS1219E

MSNAME

Stops the sending of all messages (primary requests) from a terminal except those continuing a conversation. This includes all messages destined for remote transactions with the SYSID of the MSNAME and for remote logical terminals associated with this MSNAME.

NODE

Specifies the VTAM node to be stopped and logged off. The NODE parameter can be generic if the USER keyword is not specified and applies to nodes that already exist.

The /STOP NODE command prevents future logons until a /START NODE command is issued.

/STOP NODE without the USER keyword is supported for nodes that do not yet exist. It causes the node to be created and stopped which prevents the dynamic node from logging on. /STOP NODE without the USER keyword affects all half-sessions of the specified node.

/STOP NODE USER is valid for ISC and non-ISC nodes and users; however the user must still be allocated or signed on to the node. /STOP NODE USER for ISC nodes stops the named half-session defined in USER username for NODE nodename.

/STOP NODE resets preset mode, test mode, response mode, lock node, lock lterm, pstop lterm, and purge lterm because these statuses are not significant and therefore are not kept after a logon or restart. /STOP NODE also takes other actions depending on the recovery settings for the node:

RCVYSTSN=NO

/STOP NODE acts like a /CHANGE NODE COLDSESS command for FINANCE and SLUP nodes by setting the session status to 'cold'. /STOP NODE acts like a /QUIESCE NODE command for ISC (LU6.1) nodes by initiating the shutdown and deallocating the user for the specified node. This action

changes the session status to 'cold'. With these actions taken by the /STOP NODE command, the next session initiation request for this node is allowed to again attempt a session cold start (after a /START NODE command has been entered).

RCVYCONV=NO
/STOP NODE causes any IMS conversations (active and held) to be terminated. Any conversational message that is queued or being processed will have its output response message delivered asynchronously.

RCVYFP=NO
/STOP NODE causes Fast Path status and messages to be discarded

If global resource information is kept in Resource Manager, the /STOP NODE command sets a global stop status for the node and prevents the node from logging on anywhere in the IMSplex. If global resource information is not kept in Resource Manager, /STOP NODE creates the node, if it does not exist in an ETO environment, and sets stop status for the local node. If the node does not exist in a non-ETO environment, the /STOP NODE command is rejected.

OLDS
indicates that IMS is to stop using an OLDS log data set. olds# identifies an OLDS that is defined by JCL or a DFSMDA macro and is currently started. olds# must be 00 through 99. The stopped OLDS will be dynamically deallocated when it is no longer possible for it to be accessed for dynamic backout.

If in dual mode, both primary and secondary OLDSs are stopped. If there are only two OLDS data sets available, or if the specified OLDS is the one currently being used for output, the /STOP OLDS command will be rejected.

OTMA
Causes IMS to leave the XCF group for IMS Open Transaction Manager Access (OTMA).

/STOP OTMA command processing is as follows:

1. IMS leaves the XCF group.
2. For any IMS OTMA output awaiting an ACK message, IMS aborts the message. For Commit-then-Send transactions, the output remains enqueued to the transaction pipe. For Send-then-Commit transactions, IMS aborts the transaction.

PROGRAM
Specifies the application program that is to be stopped.

/STOP PROGRAM does not stop CPI Communications driven transaction programs.

REGION
Is used to stop IMS regions and, application programs, or both. /STOP REGION is not mirrored on the XRF alternate system. You must enter this command on the alternate system if you want it to affect the alternate system.

REGION *reg#*
Is used to terminate one or more message processing regions at the conclusion of processing the current transaction. The region identifier is *reg#*.

/STOP REGION *reg#* can also be used to terminate Fast Path regions. /STOP REGION *reg#* cannot be used to terminate batch regions.

A Fast Path utility region is terminated at the next system checkpoint of the utility.

REGION *reg#–reg#*
Is used to terminate a range of message processing regions at the conclusion of processing the current transaction.

REGION *reg#* **ABDUMP** *tranname*
Causes abnormal termination of an application program.

The region identifier is *reg#* and the transaction code is *tranname*.

If the transaction indicated by *tranname* is currently running in REGION *reg#*, an error message is received at the master terminal, indicating an application program abend. The region is still be active but the transaction is stopped. The command is ignored if the transaction is not currently scheduled in region *reg#*.

`/STOP REGION reg# ABDUMP` should be used only for a region that appears to be looping or in a wait state. If this command does not abnormally terminate the application running in a region, then the `/STOP REGION reg# CANCEL` command can be used. This might cause the control region to terminate with abend 113 if parallel DL/I is being used. See `/STOP REGION reg# CANCEL` for further warnings.

The *tranname* variable is not valid for batch, IFP, or Fast Path utility regions.

REGION *reg#* **TRANSACTION** *tranname*
Stops a message processing program in wait-for-input (WFI) mode from processing within the specified region.

The region identifier is *reg#* and the transaction code is *tranname*.

If the transaction indicated by *tranname* is currently running in region *reg#*, the IMS message `DFS0569I` is received at the master terminal, indicating that a QC status code (no more messages) was returned to the application program (MPPs). The region that contained the application is still active and the transaction is not stopped. A batch message processing program in WFI mode must be stopped using the `/PSTOP` command.

A batch message processing region (BMP) scheduled against wait-for-input (WFI) transactions returns a QC status code (no more messages) for, `/PSTOP REGION`, `/DBD`, `/DBR`, or `/STA` commands only.

REGION *reg#* **CANCEL**
Is used if the region cannot be stopped with a `/STOP REGION ABDUMP` command and must be preceded by a `/STOP REGION ABDUMP` command.

The region identifier is *reg#*.

Using the `/STOP REGION CANCEL` command can cause the IMS control region to terminate with user abend 113 if parallel DL/I is being used. A z/OS CANCEL command will be rejected.

If the `/CHECKPOINT` command cannot shut down IMS or the resources are permanently unavailable, start the same type of region under the same initiator as the canceled region. This should enable IMS to detect the region as canceled and free the allocated resources.

REGION JOBNAME
Identifies regions to be stopped by their job names. The job name must be

1-8 alphanumeric or national ($,#,@) characters. The first character of the job name must be either alphabetic or national.

RTCODE
Specifies that transactions associated with this routing code are not processed.

SB
Disallows further use of sequential buffering. /STOP SB does not affect sequential buffering applications scheduled before this command was issued.

SERVGRP
Stops communications between the service group in an RSR complex at which the command was entered and the service group at the other site. /STOP SERVGRP also severs the relationship between the IMS subsystem and the TMS subsystem.

Once communications are stopped, the logger stops sending log data to the RSR tracking subsystem. No more attempts to re-establish failed conversations are made at OLDS switch. /STOP SERVGRP is normally not needed. /STOP SERVGRP is valid from an active subsystem and a tracking subsystem.

Successful completion of the syntax checking of the /STOP SERVGRP command results in the DFS058 STOP COMMAND COMPLETED message, although processing of the command continues asynchronously.

SLDSREAD
Indicates whether IMS is enabled to retrieve records from both a system log data set (SLDS) and OLDS or OLDS only. The default is that SLDSREAD is enabled.

SUBSYS
Specifies the name of the external subsystem whose connection is to be terminated. /STOP SUBSYS does allow application programs currently accessing external resources to complete normally. When those applications have terminated, the connection to the subsystem will also terminate. The application must complete all message processing before actual connection termination. The next occurrence of an external subsystem call will receive a nonzero return code, indicating the connection is not available. A /START command is then necessary to reestablish the connection.

/STOP SUBSYS can also be used to dynamically reconfigure existing subsystem definitions. The operator can issue the /STOP SUBSYS command, change or add to the PROCLIB member, and then issue the /START SUBSYS command. IMS attempts to connect those subsystems defined in the PROCLIB member.

If system failure occurs after /STOP SUBSYS is processed, the stopped status is still set.

SURVEILLANCE
Is used in an XRF environment to stop the operation of the IMS surveillance function. The SURVEILLANCE keyword parameters are:

LNK	ISC link
RDS	Restart data set
LOG	System log.
ALL	The same as specifying LNK, RDS, and LOG. This is the default.

THREAD
Stops an inactive CCTL thread. The DEDB utility region is terminated at the next system checkpoint.

The /STOP THREAD command is not valid for:

- Active CCTL threads
- BMPs

THREAD ABDUMP
Abends BMPs and DEDB utilities. If this command is used with CCTL threads, a U0474 abend results.

TMEMBER
Causes IMS to send an Open Transaction Manager Access (OTMA) command to OTMA clients to request that input be suspended for the specified transaction pipe name. IMS then stops sending output to the OTMA client and prevents any further output from being sent to the client.

TPIPE
When used with the /STOP command, TPIPE causes IMS to suspend sending output to OTMA clients.

While processing the /STOP TMEMBER TPIPE command, IMS creates a temporary transaction pipe (if one does not already exist) with the stopped status. IMS sets the synchronization status for this transaction pipe when it sends or receives the first message for the transaction pipe.

After a /STOP TMEMBER *xxx* TPIPE ALL command is issued, newly created tpipes will not be stopped for either input or output.

TRANSACTION
Stops the queuing and scheduling of messages destined for a transaction or class of transactions, or stops transaction scheduling by class. However, output can still be queued if it originates from the application program.

If the region is already scheduled and waiting for the next message (wait-for-input mode) when the command is entered, a QC status (no more messages) is returned to the application (MPPs). If there are no more messages available for the region to process, the region is not allowed to wait for the next message. Instead, a QC status is returned to the application (MPPs).

A batch message processing region (BMP) scheduled against wait-for-input (WFI) transactions returns a QC status code (no more messages) for /PSTOP REGION, /DBD, /DBR, or /STA commands only.

In a shared-queues environment, if you issue a /STOP TRANSACTION command for a transaction that is not defined on that IMS subsystem, IMS creates an SMB if the Output Creation user exit routine indicates the destination is a valid transaction. The SMB is marked as “dynamic”.

A dynamic SMB created by a /STOP TRANSACTION command can only be used to queue messages for the transaction and place the messages on the shared queues. The transaction cannot be scheduled or assigned. IMS does process checkpoints for the transaction, but does not save them across an IMS restart if they do not have a valid status.

The TRANSACTION parameter can be generic, when the generic parameter specifies a transaction that already exists.

USER
Requires the ISC user to stop or the signed on user to stop and sign off. The USER parameter can be generic and applies only to users that already exist.

For ISC users, /STOP USER specifies the ISC user that is to be made unavailable for allocation until a /START USER command is issued.

For signed on users, the /STOP USER command prevents future signons until a /START USER command is issued.

/STOP USER is supported for users that do not yet exist. It causes the user to be created and stopped, which prevents the dynamic user from signing on.

/STOP USER will not cause the user to be signed off if the associated node is not active or the associated node is not in session.

The /STOP USER command for an ETO user session resets status that is not significant such as preset mode, test mode, response mode, lock lterm, pstop lterm, and purge lterm.

/STOP USER for ETO users also takes other actions depending on the recovery settings for the user:

RCVYCONV=NO
/STOP USER causes any IMS conversations (active and held) for an ETO user to be terminated. Any conversational message that is queued or being processed will have its output response message delivered asynchronously.

RCVYFP=NO
/STOP USER causes Fast Path status and messages for an ETO user to be discarded.

If global resource information is kept in Resource Manager, the /STOP USER command sets a global stop signon status for the user and prevents the user from signing on anywhere in the IMSplex. If global resource information is not kept in Resource Manager, /STOP USER creates the user, if it does not exist in an ETO environment, and sets stop status for the local user. If the user does not exist in a non-ETO environment, the /STOP USER command is rejected.

VGRS
Causes the IMS subsystem to drop out of a generic resources group. This command is rejected if the VTAM ACB is closed (usually the result of a /STOP DC command).

While this command prevents VTAM from routing new sessions using a generic resource name to the IMS subsystem, it does not affect existing sessions, and affinities remain (until terminated through normal processing).

WADS
Indicates that a WADS is to be removed from the pool of available WADS. IMS does not allow the active WADS (if WADS mode is single), or the active WADS pair (if WADS mode is dual), to be stopped. wads# must be 0 through 9.

XRCTRACK
Results in calls to the log router to initiate or terminate XRC tracking. It is only valid on a tracking IMS system.

Examples

Example 1 for /STOP Command

Entry ET:

```
/STOP AREA DB1AREA0 DB1AREA1
```

Response ET:

```
DFS058I  STOP COMMAND IN PROGRESS
DFS0488I  STOP COMMAND COMPLETED.  AREA=DB1AREA0
DFS0488I  STOP COMMAND COMPLETED.  AREA=DB1AREA1
```

Explanation: The DEDB areas DB1AREA0 and DB1AREA1 are stopped for processing.

Example 2 for /STOP Command

Entry ET:

```
/STOP AUTOARCH
```

Response ET:

```
DFS058I  STOP COMMAND COMPLETED
```

Explanation: Automatic archiving is stopped.

Example 3 for /STOP Command

Entry ET:

```
/STOP CLASS 3
```

Response ET:

```
DFS058I  STOP COMMAND COMPLETED
```

Explanation: No further scheduling of application programs for class 3 transactions occurs. All message processing programs currently handling class 3 transactions are allowed to run until the processing limit count is reached or the input queue contains no more messages.

Example 4 for /STOP Command

Entry ET:

```
/STOP DATABASE TREEFARM
```

Response ET:

```
DFS058I  STOP COMMAND IN PROGRESS
DFS0488I  STOP COMMAND COMPLETED.  DBN=TREEFARM RC=0
```

Explanation: Database TREEFARM is stopped.

Example 5 for /STOP Command

TSO SPOC input:

```
STO DB BANKATMS BANKTERM BANKLDGR BE3ORDER
```

TSO SPOC output:

```
SYS3     DFS0488I  STO COMMAND COMPLETED. DBN= BANKATMS RC= 0
SYS3     DFS0488I  STO COMMAND COMPLETED. DBN= BANKTERM RC= 0
SYS3     DFS0488I  STO COMMAND COMPLETED. DBN= BANKLDGR RC= 0
SYS3     DFS0488I  STO COMMAND COMPLETED. DBN= BE3ORDER RC= 0
IMS3     DFS0488I  STO COMMAND COMPLETED. DBN= BANKATMS RC= 0
IMS3     DFS0488I  STO COMMAND COMPLETED. DBN= BANKTERM RC= 0
IMS3     DFS0488I  STO COMMAND COMPLETED. DBN= BANKLDGR RC= 0
IMS3     DFS0488I  STO COMMAND COMPLETED. DBN= BE3ORDER RC= 0
```

OM API input:

```
CMD (  STO DB BANKATMS BANKTERM BANKLDGR BE3ORDER )
```

OM API output:

```
<?xml version="1.0"?>
<!DOCTYPE imsout SYSTEM "imsout.dtd">
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2002.197 22:05:21.270547</statime>
<stotime>2002.197 22:05:21.307712</stotime>
<staseq>B7EFC16AF6B13F26</staseq>
<stoseq>B7EFC16AFFC40D8C</stoseq>
<rqsttkn1>USRT005 10150521</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<master>SYS3    </master>
<userid>USRT005 </userid>
<verb>STO </verb>
<kwd>DB      </kwd>
<input>STO DB BANKATMS BANKTERM BANKLDGR BE3ORDER  </input>
</cmd>
<msgdata>
<mbr name="SYS3    ">
<msg>DFS0488I  STO COMMAND COMPLETED. DBN= BANKATMS RC= 0</msg>
<msg>DFS0488I  STO COMMAND COMPLETED. DBN= BANKTERM RC= 0</msg>
<msg>DFS0488I  STO COMMAND COMPLETED. DBN= BANKLDGR RC= 0</msg>
<msg>DFS0488I  STO COMMAND COMPLETED. DBN= BE3ORDER RC= 0</msg>
</mbr>
<mbr name="IMS3    ">
<msg>DFS0488I  STO COMMAND COMPLETED. DBN= BANKATMS RC= 0</msg>
<msg>DFS0488I  STO COMMAND COMPLETED. DBN= BANKTERM RC= 0</msg>
<msg>DFS0488I  STO COMMAND COMPLETED. DBN= BANKLDGR RC= 0</msg>
<msg>DFS0488I  STO COMMAND COMPLETED. DBN= BE3ORDER RC= 0</msg>
</mbr>
</msgdata>
</imsout>
```

Explanation: The STOP command is routed from OM to the two active IMSs - SYS3 and IMS3. The response from both IMSs is returned to OM. The databases BANKATMS, BANKTERM, BANKLDGR, and BE3ORDER are stopped at both IMSs.

Example 6 for /STOP Command

Entry ET:

```
  /STOP LINE 4,5,6,7,8,9,10,11
```

Response ET:

```
  DFS058I  STOP COMMAND COMPLETED
```

Response RT:

```
DFS059I  TERMINAL STOPPED
```

Explanation: Lines 4, 5, 6, 7, 8, 9, 10, and 11 and their associated physical terminals are stopped.

Example 7 for /STOP Command

Entry ET:

```
/STOP LINE 4 8 900
```

Response ET:

```
DFS058I  STOP COMMAND COMPLETED EXCEPT LINE 900
```

Response RT:

```
DFS059I  TERMINAL STOPPED
```

Explanation: Lines 4 and 8 and their associated physical terminals are stopped. 900 is an invalid line number.

Example 8 for /STOP Command

Entry ET:

```
/STOP LINE 4 PTERM 1, 2
```

Response ET:

```
DFS058I  STOP COMMAND COMPLETED
```

Response RT:

```
DFS059I  TERMINAL STOPPED
```

Explanation: Physical terminals 1 and 2 on line 4 are stopped.

Example 9 for /STOP Command

Entry ET:

```
/STOP LTERM APPLE, TREE, FRUIT
```

Response ET:

```
DFS058I  STOP COMMAND COMPLETED
```

Response RT:

```
DFS059I  TERMINAL STOPPED
```

Explanation: Logical terminals APPLE, TREE, and FRUIT are stopped.

Example 10 for /STOP Command

Entry ET:

```
/STOP MSNAME BOSTON
```

Response ET:

```
DFS058I  STOP COMMAND COMPLETED
```

Explanation: The logical link path associated with the name BOSTON is stopped.

Example 11 for /STOP Command

Entry ET:

```
/STOP NODE HARRY
```

Response ET:

```
DFS058I  STOP COMMAND COMPLETED
```

Explanation: The physical terminal associated with node HARRY is disconnected (/CLSDST) and further logons are prevented.

Example 12 for /STOP Command

Entry ET:

```
/STO OTMA
```

Response ET:

```
DFS2361I 14:02:05 XCF GROUP CLOSED SUCCESSFULLY. SYS3
DFS058I 14:02:06 STOP COMMAND COMPLETED   SYS3
DFS996I *IMS READY*  SYS3
```

Example 13 for /STOP Command

Entry ET:

```
/STOP OLDS 09
```

Response ET:

```
DFS058I  STOP COMMAND IN PROGRESS
DFS2500I DATASET DFSOLP09 SUCCESSFULLY DEALLOCATED
DFS3257I OLDS DEALLOCATED ON DFSOLP09
```

Explanation: The subject OLDS, DFSOLP09 (DFSOLS09), will be stopped.

Example 14 for /STOP Command

Entry ET:

```
/STOP PROGRAM APPLETRE
```

Response ET:

```
DFS058I  STOP COMMAND COMPLETED
```

Explanation: Application program APPLETRE is stopped.

Example 15 for /STOP Command

Entry ET:

```
/DISPLAY A
```

Response ET:

```
REGID JOBNAME   TYPE  TRAN/STEP PROGRAM  STATUS          CLASS
    2 MPP       TP    TXCDRN24  DDLTRN24                  1,  2
    1 BMP       BMP   BMP       BMP255
    3 IFPN      FPM   NO MSG.   DDLTJN26
      DBR1CT13  DBRC
VTAM ACB CLOSED
LINE ACTIVE-IN -   1 ACTIV-OUT -   0
NODE ACTIVE-IN -   0 ACTIV-OUT -   0
LINK ACTIVE-IN -   0 ACTIV-OUT -   0
*89041/142004*
```

Explanation: Fast Path message-driven region 3 currently has no messages to process.

Entry ET:

```
/STOP REG 3
```

Response ET:

```
DFS058I STOP COMMAND IN PROGRESS
```

Entry ET:

```
/DISPLAY A
```

Response ET:

```
REGID JOBNAME    TYPE  TRAN/STEP PROGRAM  STATUS         CLASS
    2 MPP        TP    TXCDRN24  DDLTRN24                 1,   2
    1 BMP        BMP   BMP       BMP255
      FPRGN      FP    NONE
      DBRC1CT13  DBRC
VTAM ACB CLOSED
LINE ACTIVE-IN -    1 ACTIV-OUT -   0
NODE ACTIVE-IN -    0 ACTIV-OUT -   0
LINK ACTIVE-In -    0 ACTIV-OUT -   0
*89041/142102*
```

Explanation: Fast Path region 3 has been stopped. If region 3 had been processing a message, IMS would have terminated the region on completion of the transaction.

Example 16 for /STOP Command

Entry ET:

```
/DISPLAY A
```

Response ET:

```
REGID JOBNAME   TYPE  TRAN/STEP PROGRAM  STATUS         CLASS
    2 MPP       TP    TXCDRN24  DDLTRN24                 1,   2
    1 BMP       BMP   BMP       BMP255
      FPRGN     FP    NONE
      DBR1CT13  DBRC
VTAM ACB CLOSED
LINE ACTIVE-IN -   1 ACTIV-OUT -   0
NODE ACTIVE-IN -   0 ACTIV-OUT -   0
LINK ACTIVE-IN -   0 ACTIV-OUT -   0
*89041/142102*
```

Entry ET:

```
/STOP REG 1
```

Response ET:

```
DFS058I STOP COMMAND IN PROGRESS
DFS0557I STOP REGION ID NOT VALID- REGION 0001 IS BMP.
```

Explanation: `/STOP REGION` (with no keywords) is not valid for batch regions.

Example 17 for /STOP Command

Entry ET:

```
/DISPLAY A
```

Response ET:

```
REGID JOBNAME   TYPE  TRAN/STEP PROGRAM  STATUS           CLASS
    2 MPP       TP    TXCDRN24  DDLTRN24                   1,   2
    1 BMP       BMP   BMP       BMP255
    3 FPU       FPU   IFP       DBF#FPU0
      DBR1CT13  DBRC
VTAM ACB CLOSED
LINE ACTIVE-IN -   1 ACTIV-OUT -   0
NODE ACTIVE-IN -   0 ACTIV-OUT -   0
LINK ACTIVE-IN -   0 ACTIV-OUT -   0
*89041/142453*
```

Entry ET:

```
/STOP REG 3
```

Response ET:

```
DFS058I STOP COMMAND IN PROGRESS
```

Entry ET:

```
/DISPLAY A
```

Response ET:

```
REGID JOBNAME   TYPE  TRAN/STEP PROGRAM  STATUS           CLASS
    2 MPP       TP    TXCDRN24  DDLTRN24                   1,   2
    1 BMP       BMP   BMP       BMP255
      FPRGN     FP    NONE
      DBR1CT13  DBRC
VTAM ACB CLOSED
LINE ACTIVE-IN -   1 ACTIV-OUT -   0
NODE ACTIVE-IN -   0 ACTIV-OUT -   0
LINK ACTIVE-In -   0 ACTIV-OUT -   0
*89041/142758*
```

Explanation: Fast Path utility region 3 has been stopped. The Fast Path utility DBF#FPU0 was terminated at the next system checkpoint.

Example 18 for /STOP Command

Entry ET:

```
/DISPLAY A
```

Response ET:

```
REGID JOBNAME   TYPE  TRAN/STEP PROGRAM  STATUS           CLASS
    2 MPP       TP    TXCDRN24  DDLTRN24                   1,   2
    1 BMP       BMP   BMP       BMP255
      FPRGN     FP    NONE   SYS3
VTAM ACB CLOSED
LINE ACTIVE-IN -   1 ACTIV-OUT -   0
NODE ACTIVE-IN -   0 ACTIV-OUT -   0
LINK ACTIVE-IN -   0 ACTIV-OUT -   0
*89041/142758*
```

Explanation: Transaction TXCDRN24 in region 2 is looping or in a wait state.

Entry ET:

```
/STOP REG 2 ABDUMP TXCDRN24
```

Response ET:

```
DFS058I STOP COMMAND IN PROGRESS
DFS555I TRAN TXCDRN24 ABEND S000,U0474 SYS ID 220  MSG IN PROGRESS
```

Explanation: The application program has been terminated with a U0474 ABEND. This abend indicates termination in response to a user request (/STOP REGION ABDUMP).

Entry ET:

```
/DISPLAY A
```

Response ET:

```
REGID JOBNAME   TYPE  TRAN/STEP PROGRAM  STATUS         CLASS
    2 MPP       TP    WAITING                            1,   2
    1 BMP       BMP   BMP       BMP255
      FPRGN     FP    NONE
      DBR1CT13  DBRC
VTAM ACB CLOSED
LINE ACTIVE-IN -   1 ACTIV-OUT -   0
NODE ACTIVE-IN -   0 ACTIV-OUT -   0
LINK ACTIVE-IN -   0 ACTIV-OUT -   0
*89041/143420*
```

Explanation: The application has been terminated but the region remains active.

Entry ET:

```
/DISPLAY PROG DDLTRN24
```

Response ET:

```
PROGRAM   TRAN       TYPE
DDLTRN24   TXCDRN24   TP
*90340/143749*
```

Explanation: The program has not been stopped.

Entry ET:

```
/DISPLAY TRANSACTION TXCDRN24
```

Response ET:

```
   TRAN   CLS ENQCT   QCT   LCT  PLCT CP NP LP SEGSZ SEGNO PARLM RC
TXCDRN24   2     1     0 65535 65535  1  1  1     0     0     0  0
    PSBNAME: DDLTRN24
    STATUS: STOP
*90340/143802*
```

Explanation: The transaction has been stopped.

Example 19 for /STOP Command

Entry ET:

```
/DISPLAY A
```

Response ET:

```
REGID JOBNAME   TYPE  TRAN/STEP PROGRAM  STATUS           CLASS
    2 MPP       TP    WAITING                              1,   2
    1 BMP       BMP   BMP       BMP255
    3 FPU       FPU   IFP       DBP#FPU0
      DBR1CT13  DBRC
VTAM ACB CLOSED
LINE ACTIVE-IN -   1 ACTIV-OUT -   0
NODE ACTIVE-IN -   0 ACTIV-OUT -   0
LINK ACTIVE-IN -   0 ACTIV-OUT -   0
*89041/144248*
```

Entry ET:

```
/STOP REG 3 ABDUMP
```

Response ET:

```
DFS058I STOP COMMAND IN PROGRESS
```

Explanation: A transaction code is not entered when terminating a Fast Path utility with a /STOP REGION ABDUMP command.

Example 20 for /STOP Command

Entry ET:

```
/DISPLAY A
```

Response ET:

```
REGID JOBNAME   TYPE  TRAN/STEP PROGRAM  STATUS            CLASS
    2 MPP       TP    TXCDRN24  DDLTRN24 WAIT-INPUT          1,   2
    1 BMP       BMP   BMP       BMP255
      FPRGN     FP    NONE
      DBR1CT13  DBRC
VTAM ACB CLOSED
LINE ACTIVE-IN -   1 ACTIV-OUT -   0
NODE ACTIVE-IN -   0 ACTIV-OUT -   0
LINE ACTIVE-IN -   0 ACTIV-OUT -   0
*89041/150141*
```

Explanation: Message processing program DDLTRN24 is waiting for an input message.

Entry ET:

```
/STOP REGION 2 TRANSACTION TXCDRN24
```

Response ET:

```
DFS058I STOP COMMAND IN PROGRESS
DFS0569I PSTOP OR STOP COMPLETE FOR REGION0002 TRAN TXCDRN24.
```

Explanation: A QC status code was returned to the WFI application program DDLTRN24.

Entry ET:

```
/DISPLAY A
```

Response ET:

```
REGID JOBNAME   TYPE  TRAN/STEP PROGRAM  STATUS         CLASS
    2 MPP       TP    WAITING                           1,   2
    1 BMP       BMP   BMP       BMP255
      FPRGN     FP    NONE
      DBR1CT13  DBRC
VTAM ACB CLOSED
LINE ACTIVE-IN -   1 ACTIV-OUT -   0
NODE ACTIVE-IN -   0 ACTIV-OUT -   0
LINK ACTIVE-In -   0 ACTIV-OUT -   0
*89041/150206*
```

Explanation: The WFI application has been terminated but the region is still active.

Entry ET:

```
/DISPLAY TRANSACTION TXCDRN24
```

Response ET:

```
   TRAN   CLS ENQCT   QCT   LCT  PLCT CP NP LP SEGSZ SEGNO PARLM RC
TXCDRN24   2     4     0 65535 65535  1  1  1     0     0     0  0
    PSBNAME: DDLTRN24
*90340/150219*
```

Explanation: The transaction is not stopped.

Example 21 for /STOP Command

Entry ET:

```
/DISPLAY A
```

Response ET:

```
REGID JOBNAME   TYPE  TRAN/STEP PROGRAM  STATUS            CLASS
    2 MPP       TP    TXCDRN24  DDLTRN24 WAIT-INPUT          1,   2
    1 BMP       BMP   BMP       BMP255
      FPRGN     FP    NONE
      DBR1CT13  DBRC
VTAM ACB CLOSED
LINE ACTIVE-IN -   1 ACTIV-OUT -   0
NODE ACTIVE-IN -   0 ACTIV-OUT -   0
LINK ACTIVE-IN -   0 ACTIV-OUT -   0
*89041/150813*
```

Entry ET:

```
/STOP REGION 2 TRANSACTION TRAN255
```

Response ET:

```
DFS058I STOP COMMAND IN PROGRESS
DFS0558I TRAN TRAN255 NOT SCHEDULED
```

Explanation: TRAN255 is a valid transaction for the IMS system but it is not currently scheduled in region 2. If TRAN255 had not been a valid transaction for the IMS system, only message `DFS230I (TRAN SPECIFIED WITH ABDUMP OR TRAN KEYWORD IS NOT VALID)` would have been issued.

Example 22 for /STOP Command

Entry ET:

```
D A,L
```

Response ET:

```
 JOBS     M/S    TS USERS    SYSAS    INITS   ACTIVE/MAX VTAM     OAS
00001    00010    00001      00019    00020    00001/00020       00000
 LLA      LLA      LLA      NSW  S  VLF      VLF      VLF      NSW  S
 JES2     JES2     IEFPROC  NSW  S  RMF      RMF      IEFPROC  NSW  S
 IMSVTAM  IMSVTAM  IEFPROC  NSW  S  TSO      TSO      STEP1    NSW  S
 CQS      CQS      IEFPROC  NSW  S  IMSECTA9 IMSECTA9 IEFPROC  NSW  S
 DLIECTA9 DLIECTA9 DLISAS   NSW  S  DBRECTA9 DBRECTA9 DBRC     NSW  S
 MPP610C  MPP      MPP      NSW  J
 USRT001 OWT
```

Explanation: MPP610C is an IMS message processing region.

Entry ET:

```
/STOP REGION JOBNAME MPP610C
```

Response ET:

```
DFS058I STOP COMMAND IN PROGRESS
DFS552I MESSAGE REGION MPP610C  STOPPED ID=00001 TIME=1616  SYSX

SMF000I  MPP610C     MPP         DFSRRC00    0000
$HASP395 MPP610C  ENDED
```

Example 23 for /STOP Command

Entry ET:

```
/STOP SB
```

Response ET:

```
DFS058  STOP COMMAND COMPLETED
```

Entry ET:

```
/DISPLAY POOL DBAS
```

Response ET:

```
SEQUENTIAL BUFFERING:  STATUS = STOPPED
  MAX      N.A.  FREE   N.A.   CURR     0K  HIGH  320K
DATABASE BUFFER POOL: SIZE   67584
  REQ1       0 REQ2     0 READ     0 BISAM     0 WRITES     0
  KEYC       0 LCYL     0 PURG     0 OWNRR     0 ERRORS 00/00
  DATABASE BUFFER POOL:  BSIZE  12288
    RRBA     0 RKEY     0  BFALT    0 NREC  0 SYN PTS     0
    NMBUFS  29 VRDS     0   FOUND   0 VWTS  0  ERRORS 00/00
  DATABASE BUFFER POOL:  BSIZE  356352
    RRBA     0 RKEY     0  BFALT    0 NREC  0 SYN PTS     0
    NMBUFS  29 VRDS     0  FOUND    0 VWTS  0  ERRORS 00/00
  *90253/104547*
```

Explanation: Sequential buffering is stopped.

Example 24 for /STOP Command

Entry ET:

```
/STOP SUBSYS ALL
```

Response ET:

```
DFS058I  STOP COMMAND IN PROGRESS
```

Explanation: IMS has initiated the termination of the connection. When all dependent regions have terminated their connections, IMS will complete the termination. It is likely that an external subsystem message indicating connection termination will be received at this time.

Example 25 for /STOP Command

Entry ET:

```
/STOP SUBSYS XXX1 XXX3
```

Response ET:

```
DFS058I  STOP COMMAND IN PROGRESS
```

Explanation: IMS has initiated the termination of the connection. When all dependent regions have terminated their connections, IMS will complete the termination. It is likely that an external subsystem message indicating connection termination will be received at this time

Example 26 for /STOP Command

Entry ET:

```
/DISPLAY A THREAD
```

Response ET:

```
REGID JOBNAME   TYPE  TRAN/STEP PROGRAM  STATUS     CLASS
      BATCHREG  BMP   NONE
      FPRGN     FP    NONE
    2 CICS1A    DBT   IEFPROC   BMP255   ACTIVE
    3 CICS1A    DBT   IEFPROC   PLVAPZ12 ACTIVE
    1 CICS1A    DBT   IEPROC              AVAILABLE
      DBRCHTA1  DBRC
      DLICHTA1  DLS
*00082/142907*
```

Entry ET:

```
/STOP THREAD 2
```

Response ET:

```
DFS058I STOP COMMAND IN PROGRESS
DFS0556I COMMAND REJECTED; DBCTL THREAD IS ACTIVE
```

Example 27 for /STOP Command

Entry ET:

```
/DISPLAY A THREAD
```

Response ET:

```
REGID JOBNAME   TYPE  TRAN/STEP PROGRAM  STATUS     CLASS
      BATCHREG  BMP   NONE
      FPRGN     FP    NONE
    2 CICS1A    DBT   IEFPROC   BMP255   ACTIVE
    3 CICS1A    DBT   IEFPROC   PLVAPZ12 ACTIVE
    1 CICS1A    DBT   IEFPROC              AVAILABLE
      DBRCHTA1  DBRC
      DLICHTA1  DLS
*00082/143027*
```

Entry ET:

```
/STOP THREAD 1
```

Response ET:

```
DFS058I STOP COMMAND IN PROGRESS
```

Entry ET:

```
/DISPLAY A THREAD
```

Response ET:

```
REGID JOBNAME   TYPE  TRAN/STEP PROGRAM  STATUS     CLASS
      BATCHREG  BMP   NONE
      FPRGN     FP    NONE
    2 CICS1A    DBT   IEFPROC   BMP255   ACTIVE
    3 CICS1A    DBT   IEFPROC   PLVAPZ12 ACTIVE
      DBRCHTA1  DBRC
      DLICHTA1  DLS
*00082/143055*
```

Example 28 for /STOP Command

Entry ET:

```
/DISPLAY A THREAD
```

Response ET:

```
REGID JOBNAME   TYPE  TRAN/STEP PROGRAM  STATUS     CLASS
      BATCHREG  BMP   NONE
      FPRGN     FP    NONE
    2 CICS1A    DBT   IEFPROC   BMP255   ACTIVE
    3 CICS1A    DBT   IEFPROC   PLVAPZ12 ACTIVE
      DBRCHTA1  DBRC
      DLICHTA1  DLS
*00082/144731*
```

Entry ET:

```
/STOP THREAD 2 ABDUMP
```

Response ET:

```
/DFS058I STOP COMMAND IN PROGRESS
```

Response ET:

```
DFS554A CICS1A     00002 IEFPROC  BMP255   (3)        000,0474   20
/082 14:49:11    RTKN= CICS1   B3C81CB789F4BE83
```

Entry ET:

```
/DISPALY A THREAD
```

Response ET:

```
REGID JOBNAME   TYPE  TRAN/STEP PROGRAM  STATUS     CLASS
      BATCHREG  BMP   NONE
      FPRGN     FP    NONE
    3 CICS1A    DBT   IEFPROC   PLVAPZ12 ACTIVE
      DBRCHTA1  DBRC
      DLICHTA1  DLS
*00082/145038*
```

Example 29 for /STOP Command

Entry ET:

```
/STO TMEMBER CLIENT1 TPIPE TPIPESY
```

Response ET:

```
DFS058I 15:38:03 STOP COMMAND COMPLETED   SYS3
DFS996I *IMS READY*  SYS3
```

Example 30 for /STOP Command

Entry ET:

```
  /STOP TRANSACTION ALL CLASS 6
```

Response ET:

```
  DFS058I  STOP COMMAND COMPLETED
```

Explanation: All transactions associated with class 6 will be marked as stopped and all class 6 transactions are no longer available for scheduling. All message

processing regions currently processing class 6 transactions are allowed to run until the processing limit count is reached or the input queue contains no more messages.

Example 31 for /STOP Command

Entry ET:

```
/STOP TRANSACTION PIT, SEED
```

Response ET:

```
DFS058I  STOP COMMAND COMPLETED
```

Explanation: Transaction codes PIT and SEED are stopped.

Example 32 for /STOP Command

Entry ET:

```
/DISPLAY USER IMS*
```

Response ET:

```
USER     ENQCT DEQCT   QCT   SYS3
IMSUS06      0     0     0 ALLOC(DTSLU602)
IMSUS04      0     0     0 ALLOC(DTSLU603)
IMSUS03      0     0     0 ALLOC(DTSLU601)
IMSUS02      0     0     0 ALLOC(DTSLU202)
IMSUS01      0     0     0 ALLOC(DTSLU201)
IMSUS09    N/A   N/A   N/A ALLOC(ENDS02  ) STATIC
IMSUS08    N/A   N/A   N/A ALLOC(ENDS01  ) STATIC
IMSUS11    N/A   N/A   N/A ALLOC(ENDS03  ) STATIC
IMSUS10    N/A   N/A   N/A ALLOC(OMSSLU2A) STATIC
*91091/111454*
```

Entry ET:

```
/STOP USER IMSUS01 IMSUS02
```

Response ET:

```
DFS058I 11:16:24 STOP COMMAND COMPLETED
```

Entry ET:

```
/STOP USER HELLO%
```

Response ET:

```
DFS3633 11:18:25 GENERIC PARAMETER RESOURCES NOT FOUND, NO ACTION TAKEN
```

Entry ET:

```
/DISPLAY USER IMSUS01 IMSUS02
```

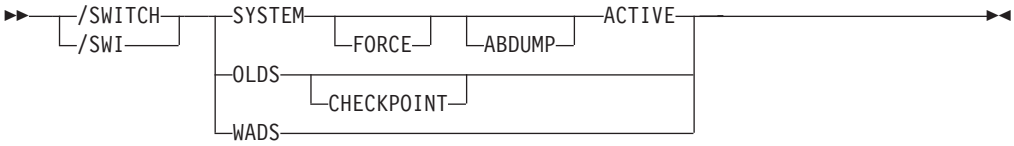
Response ET:

```
USER     ENQCT DEQCT   QCT
IMSUS01      0     0     0 STOPPED
IMSUS02      0     0     0 STOPPED
*91091/111727*
```

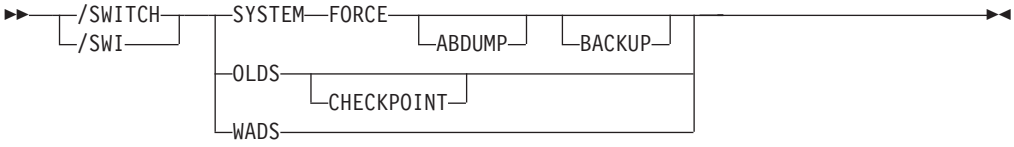
Chapter 60. /SWITCH

Format

/SWITCH for an Active XRF Subsystem



/SWITCH for an Alternate XRF Subsystem



Environments and Keywords

Table 152 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 152. Valid Environments for the /SWITCH Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/SWITCH	X	X	X
ABDUMP	X		X
ACTIVE	X		X
BACKUP	X		X
CHECKPOINT	X	X	X
FORCE	X		X
OLDS	X	X	X
SYSTEM	X		X
WADS	X	X	X

Usage

The /SWITCH command is used to switch active data sets or change between the active and alternate systems. Certain combinations of keywords are valid only in the active or alternate systems, as shown in the format in the syntax diagram. For a list of the commands recovered, see “Commands Recovered During Emergency Restart” on page 26.

SYSTEM
Requests a takeover by the alternate system from the current active system in an XRF environment.

© Copyright IBM Corp. 1974, 2003

/SWITCH SYSTEM without the FORCE keyword causes the active system to attempt to gracefully quiesce before the alternate system becomes active. System activity currently in progress is allowed to complete. New activity is queued. This disables surveillance on the active system, but not the alternate. Therefore, surveillance (if enabled) will eventually trigger a takeover if it does not eventually quiesce.

Unless the optional FORCE keyword is specified, the command is only operable when entered on the active system.

FORCE
Causes an immediate termination of the active system, forcing the alternate system to become the active.

ABDUMP
Results in a diagnostic dump of the active system when entered from either the active system or the alternate system (if it is on the same processor as the active system).

ACTIVE, BACKUP
Indicates the system on which the command is being entered. The keyword ACTIVE is required when the command is entered on an active system. The keyword BACKUP is optional when the command is entered on an alternate system. /SWITCH SYSTEM FORCE, without the ACTIVE keyword, can only be entered on an alternate system. This prevents the inadvertent abend of a newly created active system that is mistakenly assumed to still be the alternate system.

OLDS, CHECKPOINT
Causes switching of the active log data set. This log switch capability is identical to that provided with /DBDUMP and /DBRECOVERY commands. You can specify the CHECKPOINT keyword to take a simple checkpoint after the active log data set has been switched to the next OLDS. The /SWITCH OLDS CHECKPOINT command operates in all IMS environments.

WADS
Causes switching of the active write-ahead log data set. If you are using dual logging for the WADS, this command causes IMS to use the next available WADS pair.

This command is rejected if no unused WADS is available, or for dual logging, if no unused pair of WADS is available.

Examples

Example 1 for /SWITCH Command

Entry ET:

```
/SWITCH OLDS
```

Response ET:

```
DFS3257I ONLINE LOG NOW SWITCHED
DFS058I 17:10:51 SWITCH COMMAND COMPLETED
```

Entry ET:

```
/SWITCH OLDS CHECKPOINT
```

Response ET:

```
DFS3257I ONLINE LOG NOW SWITCHED - FROM DFSOLP01 TO DFSOLP02
DFS058I 17:12:53 SWITCH COMMAND COMPLETED

DFS2719I MSDB CHECKPOINT WRITTEN TO MSDBCP2
DFS994I *CHKPT 91057/171254**SIMPLE*
DFS3499I ACTIVE DDNAMES: MODBLKSA IMSACBA  FORMATA  MODSTAT ID:    1
DFS3804I LATEST RESTART CHKPT: 91057/132000, LATEST BUILDQ CHKPT:
91057/132414
```

Example 2 for /SWITCH Command

Entry ET (Master Terminal for active system IMSA):

```
/SWITCH SYSTEM FORCE
```

Response ET:

A response message is not returned for the /SWITCH SYSTEM FORCE command. Any further input to the master terminal of the active system is inhibited.

Response RT (z/OS console for active system IMSA):

The z/OS console for the active system will show a user 0604 abend in progress for IMSA. If the ABDUMP keyword had been included on the above /SWITCH command, the 0604 abend would be accompanied by a diagnostic dump of the active system.

Response RT (master terminal for alternate system IMSB):

Figure 6 is a screen that shows some of the messages associated with the beginning of takeover on the alternate system.

```
02/05/15 15:28:27  RSENAME: DFSRSENM  BACKUP   TAKEOVER IN PROGRESS  IMSB
DFS3890I 15:27:18 TAKEOVER REQUESTED
DFS970I 15:28:05 UNEXPECTED  STATUS   ,NODE APPLA  ,USER  N/A  ,SEND  ,RC
=14,FDB2=13,NSECIT =29,SENSE=00000000,REASON=00
DFS3257I ONLINE LOG CLOSED ON DFSOLP00
DFS3891I 15:28:18 TAKEOVER IN PROGRESS

 DFS2591I NO MSDB HEADERS FOUND, IMAGE COPY LOAD IGNORED
DFS3839I 14:26:46 XRF INITIAL DB PRE-OPEN COMPLETE.
DFS3838I 14:28:41 XRF INITIAL DC PRE-OPEN COMPLETE.
--------------------------------------------------------------------------------









                                                        PASSWORD:
```

Figure 6. Alternate System at Start of Takeover

Intermediate screens are not shown. They would indicate such takeover functions as:

- Enabling of dependent region processing
- IRLM takeover
- Backout processing
- Draining of suspend queue

- Session switching

Figure 7 is a screen that show takeover is complete.

```
02/05/15 15:30:59  RSENAME: DFSRSENM  ACTIVE  AWAITING I/O PREVENTION  IMSB
DFS2716I NO MSDBS FOUND - NO MSDB CHECKPOINT TAKEN
 DFS994I *CHKPT 85135/152931**SIMPLE**
DFS3499I ACTIVE DDNAMES: MODBLKSA IMSACBA  FORMATA  MODSTAT ID:   11
DFS3804I LAST CHKPT ID VALID FOR RESTART: 85135/152931-BUILDQ: 85135/142629


DFS994I TAKEOVER COMPLETED.
DFS3859I 15:29:19 PRIORITY 4 SESSIONS SWITCHED.
DFS3860I 15:29:19 ALL TERMINAL SESSIONS SWITCHED.
--------------------------------------------------------------------------------









                                                        PASSWORD:
```

Figure 7. Newly Created Active System after Takeover

Takeover is complete and the alternate system is now an active system. The XRF environment status line indicates that the newly created active system is running in I/O toleration mode (awaiting I/O prevention). For a more complete description of I/O toleration and I/O prevention, see the *IMS Version 9: Operations Guide*.

Chapter 61. TERMINATE

Format

TERMINATE OLC

Environments and Keywords

Table 153 lists the environments (DB/DC, DBCTL, and DCCTL) from which the TERMINATE command can be issued.

Table 153. Valid Environments for the TERMINATE OLC Command

Command / Keyword	DB/DC	DBCTL	DCCTL
TERMINATE OLC	x	x	x

TERMINATE OLC

When the TERMINATE OLC (stop online change) command is issued by an IMS command master that is running with RM services (RMENV=Y), the command terminates a global online change and coordinates with all of the IMSs in the IMSplex.

If an IMS is not running with RM services (RMENV=N), the TERMINATE OLC command terminates an online change for that IMS only. If an IMS system is not running with RM, each IMS must have a unique OLCSTAT data set which cannot be shared. If the OLCSTAT data set contains the name of an IMS other than the one that is processing the online change, TERMINATE OLC is rejected because the OLCSTAT data set is invalid for the environment. To determine which IMS member names are invalid, issue the QUERY OLC command to display the contents of the OLCSTAT data set. You can use the OLC utility, DFSUOLC0, to correct the data set.

Each IMS system that does not have RM services and participates in global online change, must separately issue the TERMINATE OLC command. In a no RM environment, if more than one IMS is specified in the route list for the TERMINATE OLC or the default of route all is specified, online change is only performed for the IMS command master. To determine which IMSs are defined with RMENVNO, issue a QUERY MEMBER SHOW(ATTRIB) command.

A TERMINATE OLC command that aborts a global online change is similar to the /MODIFY ABORT command, except that it applies to all of the IMSs in an IMSplex that are participating in the global online change.

This command may be specified only through the OM API. OM sends the TERMINATE OLC command to an IMS in the IMSplex.

TERMINATE OLC can be used to abort an IMSplex-wide global online change initiated by a INITIATE OLC PHASE(PREPARE) command, before the online change is successfully committed with a INITIATE OLC PHASE(COMMIT) command.

© Copyright IBM Corp. 1974, 2003

TERMINATE OLC can be used to abort an online change after an INITIATE OLC PHASE(COMMIT) failure that occurs before the OLCSTAT data set is updated. Once the commit process has updated the OLCSTAT data set, the online change is considered to be successful and cannot be aborted.

TERMINATE OLC is not supported if local online change is enabled. The TERMINATE OLC command is rejected if the IMS to which the command is routed does not support global online change. If this occurs and there is an IMS that supports global online change, the user must route the command to a specific IMS that supports global online change.

The TERMINATE OLC command is invalid on the XRF alternate, RSR tracker, and FDR system.

The OM command time-out default of 300 seconds (5 minutes) may not be enough time for the online change phase to complete. It may be required to specify a time-out value on the command based on the needs of the installation.

The command syntax for this command is defined in XML and is available to automation programs which communicate with OM.

TERMINATE OLC Output Fields

Table 154 shows the output fields for a TERMINATE OLC command. The columns in the table are as follows:

Short Label
Contains the short label generated in the XML output.

Keyword
Identifies keywords on the command that caused the field to be generated. N/A appears for output fields that are always returned.

Meaning
Provides a brief description of the output field.

Table 154. Output Fields for the TERMINATE Command

SHORT LABEL	KEYWORD	MEANING
MBR	N/A	IMSplex member that built the line of output. IMS identifier of the IMS that was master of the abort phase. IMS identifier is always returned.
IMSMBR	N/A	IMS member that performed the global online change phase. The IMS member name is always returned.
CC	N/A	Completion code from the IMS member that performed the online change phase. Completion code is always returned.
ERRT	N/A	Error text associated with a nonzero completion code returned by the IMS member that performed the online change phase. Error text may be returned if the completion code is nonzero.

Return, Reason, and Completion Codes for TERMINATE OLC

The OM return and reason codes that may be returned as a result of this command are standard for all commands entered through the OM API. Refer to the OM CSLOMCMD Return and Reason code section for the list of codes and their meanings.

An IMS return and reason code is returned to OM by the TERMINATE OLC command.

Some reason codes are accompanied by a complete list of IMSs and completion codes. The reason code meaning indicates whether a list is returned. A partial list of IMSs and completion codes may be returned with any TERMINATE OLC error reason code, if any output was built before the error was detected.

Table 155 contains the return and reason codes that can be returned to OM by the TERMINATE OLC command.

Table 155. Return and Reason Codes for the TERMINATE Command

Return Code	Reason Code	Meaning
X'00000000'	X'00000000'	The TERMINATE OLC command completed successfully. The TERMINATE OLC command is applied to all of the IMSs listed in the OLCSTAT data set. All of the IMSs in the IMSplex are no longer in an online change state. An output line is built for each IMS listed in the OLCSTAT data set. Each output line contains the IMS member name and a completion code of zero.
X'00000004'	X'0000100C'	The TERMINATE OLC command completed successfully, but was not applicable to one or more IMSs for acceptable reasons. The TERMINATE OLC command applies to all of the IMSs listed in the OLCSTAT data set. An output line is built for each IMS listed in the OLCSTAT data set. Each output line contains the IMS member name and a completion code. A nonzero completion code may be accompanied by error text. One or more of the IMSs contain a completion code indicating the terminate online change did not apply to this IMS, such as the IMS state is abended, the IMS state is shutdown, or this IMS is already in the correct online change state. The TERMINATE OLC completion code table contains the list of completion codes and error text that can be returned by the TERMINATE OLC command.
X'0000000C'	X'00003000'	The TERMINATE OLC command is successful for at least one IMS but not all IMSs. The TERMINATE OLC command applies to all IMSs listed in the OLCSTAT data set. An output line is built for each IMS listed in the OLCSTAT data set. Each output line contains the IMS member name and a completion code. A nonzero completion code may be accompanied by error text. One or more of the IMSs returned an error completion code. The TERMINATE OLC completion code table contains the list of completion codes and error text that can be returned by the TERMINATE OLC command. If the TERMINATE OLC command fails for one or more IMSs, correct the problem and issue the TERMINATE OLC command again. See 'TERMINATE OLC error handling' for more details.

Table 155. Return and Reason Codes for the TERMINATE Command (continued)

Return Code	Reason Code	Meaning
X'0000000C'	X'00003004'	The TERMINATE OLC command failed for all of the IMSs. The TERMINATE OLC command applies to all of the IMSs listed in the OLCSTAT data set. An output line is built for each IMS listed in the OLCSTAT data set. Each output line contains the IMS member name and a completion code. A nonzero completion code may be accompanied by error text. All of the IMSs returned an error completion code. The TERMINATE OLC completion code table contains the list of completion codes and error text that can be returned by the TERMINATE OLC command. If the TERMINATE OLC command fails for one or more IMSs, correct the problem and issue the TERMINATE OLC command again. See 'TERMINATE OLC error handling' for more details.
X'00000010'	X'00004004'	The TERMINATE OLC command failed because there is no CQS. RM attempted to access the process resource on the resource structure, but it failed because CQS is not available. The online change phase may have succeeded on one or more IMSs. See 'TERMINATE OLC error handling' for more details.
X'00000010'	X'0000400C'	The TERMINATE OLC command failed because it is invalid for an XRF alternate.
X'00000010'	X'00004014'	The TERMINATE OLC command failed because it is invalid for an RSR tracker.
X'00000010'	X'00004018'	The TERMINATE OLC command failed because the RM resource structure is not available. One or more IMSs in the IMSplex may still be in an online change state. See 'TERMINATE OLC error handling' for more details.
X'00000010'	X'0000401C'	The TERMINATE OLC command failed because it is invalid for an FDR region.
X'00000010'	X'00004100'	The TERMINATE OLC command is rejected because the resource structure is full. RM failed trying to create the process resource on the resource structure. One or more IMSs may still be in an online change state. See 'TERMINATE OLC error handling' for more details.
X'00000010'	X'00004104'	The TERMINATE OLC command failed because RM is not available. The online change phase may have succeeded on one or more IMSs'. Either there is no RM address space, or RM is active but not registered to SCI because CQS or the resource structure is not available. See 'TERMINATE OLC error handling' for more details.
X'00000010'	X'00004108'	The TERMINATE OLC command failed because SCI is not available. One or more IMSs may still be in an online change state. See 'TERMINATE OLC error handling' for more details.
X'00000010'	X'0000410C'	The TERMINATE OLC command is rejected, because global online change is not enabled. Local online change is enabled. Use the /MODIFY command for local online change. If your IMSplex is made up of some IMSs that support global online change and some that support local online change, route the TERMINATE OLC command to an IMS that is enabled for global online change.

Table 155. Return and Reason Codes for the TERMINATE Command (continued)

Return Code	Reason Code	Meaning
X'00000010'	X'00004110'	The TERMINATE OLC command is rejected, because the command does not apply to the online change state of the command master. TERMINATE OLC is rejected if the command master is not in an online change state. TERMINATE OLC is rejected if the command master has already committed the online change. See 'TERMINATE OLC error handling' for more details.
X'00000010'	X'0000412C'	The OLCSTAT data set contains the name of an IMS other than the IMS processing the online change. Use DFSUOLC0 to correct the data set.
X'00000010'	X'00004114'	The TERMINATE OLC command failed because of an error accessing the OLCSTAT data set. One or more IMSs in the IMSplex may still be in an online change state. A DFS2843 message is sent to the OM output exit as unsolicited output. See 'TERMINATE OLC error handling' for more details.
X'00000010'	X'00004118'	The TERMINATE OLC command failed because of an error allocating the OLCSTAT data set. One or more IMSs in the IMSplex may still be in an online change state. A DFS2848 message is sent to the OM output exit as unsolicited output. See 'TERMINATE OLC error handling' for more details.
X'00000010'	X'0000411C'	The TERMINATE OLC command failed because of an error in the OLCSTAT data set contents. One or more of the values is invalid. A DFS2844 message is sent to the OM output exit as unsolicited output.
X'00000010'	X'00004120'	The TERMINATE OLC command is rejected because an online change phase is already in progress on this IMS, which may be INITIATE OLC, TERMINATE OLC, or /DISPLAY MODIFY.
X'00000014'	X'00005000'	The TERMINATE OLC command is rejected because an IMODULE GETSTOR storage request failed.
X'00000014'	X'00005004'	The TERMINATE OLC command failed because a DFSOCMD response buffer could not be obtained. One or more IMSs in the IMSplex may still be in an online change state. See 'TERMINATE OLC error handling' for more details.
X'00000014'	X'00005100'	The TERMINATE OLC command failed because of an RM error. One or more IMSs in the IMSplex may still be in an online change state. See 'TERMINATE OLC error handling' for more details.
X'00000014'	X'00005104'	The TERMINATE OLC command failed because of a CQS error. One or more IMSs in the IMSplex may still be in an online change state. See 'TERMINATE OLC error handling' for more details.
X'00000014'	X'00005108'	The TERMINATE OLC command failed because of an SCI error. One or more IMSs in the IMSplex might still be in an online change state. See 'TERMINATE OLC error handling' for more details.

Table 155. Return and Reason Codes for the TERMINATE Command (continued)

Return Code	Reason Code	Meaning
X'00000014'	X'00005FFF'	The TERMINATE OLC command failed because of an internal IMS error. One or more IMSs in the IMSplex might still be in an online change state. See 'TERMINATE OLC error handling' for more details.

TERMINATE OLC Error Handling

Errors unique to the processing of this command are returned as a completion code. A completion code is returned for each action against an individual resource.

The TERMINATE OLC command may result in an error that leaves IMSs in various online change states. Correct the error. Issue the QUERY MEMBER TYPE(IMS) SHOW(STATUS) command to display the online change state of all the IMSs in the IMSplex. Evaluate the QUERY MEMBER TYPE(IMS) output to help you determine what to do:

- None of the IMSs in an online change state

  The TERMINATE OLC command succeeded or was not applicable. No further action needs to be taken.
- Some of the IMSs in a prepare complete state

  The online change is not committed. Correct the problem that caused TERMINATE OLC to fail, then retry the TERMINATE OLC command again.

  The IMSs that are in an online change state remain in an online change state until you abort the online change.
- All IMSs in a prepare complete state

  The online change is not committed. Correct the problem that caused TERMINATE OLC to fail, then retry the TERMINATE OLC command again.

  The IMSs that are in an online change state remain in an online change state until you abort the online change.
- Some IMSs in prepare complete and commit phase 1 complete state

  The commit phase failed before the master updated the OLCSTAT data set, so the online change is not committed. Correct the problem that caused TERMINATE OLC to fail and retry the TERMINATE OLC command.

  The IMSs that are in an online change state remain in an online change state until you abort the online change.
- All IMSs in commit phase 1 complete state.

  If the commit phase failed before the master updated the OLCSTAT data set, the online change is not committed. Correct the problem that caused TERMINATE OLC to fail and retry the TERMINATE OLC command.

  If the commit phase failed after the master updated the OLCSTAT data set, the online change is committed. The TERMINATE OLC command is not permitted. You must correct the problem that caused the commit command to fail and retry INITIATE OLC PHASE(COMMIT).

  The IMSs that are in an online change state remain in an online change state until you abort the online change or commit the online change.

  You can determine whether the OLCSTAT data set has been updated by the modify id. Issue the QUERY OLC LIBRARY(OLCSTAT) SHOW(MODID) command. Check if the modify id returned is different from the modify id returned by the INITIATE

OLC PHASE(PREPARE) command, or the modify id returned by a QUERY OLC LIBRARY(OLCSTAT) SHOW(MODID) command issued before the INITIATE OLC PHASE(COMMIT) command.

- Some IMSs in commit phase 1 complete state and some in commit phase 2 complete state

  The online change is committed. The TERMINATE OLC command is not permitted. You must correct the problem that caused the commit command to fail and retry INITIATE OLC PHASE(COMMIT).

  The IMSs that are in an online change state remain in an online change state until you finish the online change with an INITIATE OLC PHASE(COMMIT) command.
- All IMSs in commit phase 2 complete state

  The online change is committed. The TERMINATE OLC command is not permitted. You must correct the problem that caused the commit command to fail and retry INITIATE OLC PHASE(COMMIT).

  The IMSs that are in an online change state remain in an online change state until you finish the online change with an INITIATE OLC PHASE(COMMIT) command.
- Some IMSs in commit phase 2 complete state and some not in online change state.

  The online change is committed. The TERMINATE OLC command is not permitted. You must correct the problem that caused the commit command to fail and retry INITIATE OLC PHASE(COMMIT).

  The IMSs that are in an online change state remain in an online change state until you finish the online change with an INITIATE OLC PHASE(COMMIT) command.

Errors unique to the processing of this command are returned as a completion code. A completion code is returned for an IMS participating in the online change phase.

Table 156 contains the completion codes that can be returned on a TERMINATE OLC command, the meaning of the completion code, and any error text associated with the code.

Table 156. Completion Codes for the TERMINATE Command

Completion Code	Meaning	ERROR TEXT (upper case)
0	The online change commit or abort phase completed successfully.	
1	The online change type does not apply to this IMS. For example, an ACBLIB online change does not apply to a DCCTL IMS. This IMS does nothing.	

Table 156. Completion Codes for the TERMINATE Command (continued)

Completion Code	Meaning	ERROR TEXT (upper case)
2	The online change phase was not attempted by this IMS for one of the following reasons: • The online change phase master encountered an error and did not direct this IMS to perform the online change phase.	
3	This IMS is already in the correct online change state. This IMS does not have to do anything.	
2 58	An IMS is not registered to RM. An OLCSTAT dataset contains an IMS that is not registered to RM. Terminate fails for that IMS.	
60	IMODULE GETMAIN storage error.	
61	BCB storage error.	
62	HIOP storage error.	
63	WKAP storage error.	
80	Data set error.	Function (8 char), ddname (8 char), return code (8 bytes), and error detail (8 char). Function can be one of the following: • OPEN Data set open error. • READ Data set read error. DDname can be OLCSTAT. Return code is the data set service return code. Reason code is the data set service reason code.
90	Internal error.	Module name that detected internal error (8 char), unused (8 char), return code or function code (8 bytes), and error detail (8 char).

Table 156. Completion Codes for the TERMINATE Command (continued)

Completion Code	Meaning	ERROR TEXT (upper case)
91	The online change commit phase 2 or abort phase timed out before this IMS responded to the online change commit phase 2 or abort phase. The commit phase 2 or abort may have succeeded on this IMS. Issue `QUERY MEMBER TYPE(IMS)` to determine the online change state of this IMS.	

Table 156. Completion Codes for the TERMINATE Command (continued)

Completion Code	Meaning	ERROR TEXT (upper case)
B2	IMS state error.	IMS state error (32 char). The IMS state can be one of the following: • ABENDED This IMS abended since the last successful online change. Online change is terminated on this IMS. • NOT-REACHABLE The online change phase is rejected because this IMS is NOT-REACHABLE. The SCI on the OS image where this IMS is active is down. Restart the SCI and re-issue the `INITIATE OLC` or `TERMINATE OLC` command. • OLC ALREADY COMMITTED The online change terminate is rejected because online change is already committed. All IMSs have completed commit phase 1 and the OLCSTAT data set was updated. • OLC NOT IN PROGRESS The IMS is not in an online change state. The request to terminate the online change does not apply to this IMS. • OLC PHASE IN PROGRESS The online change phase is rejected because this IMS has an online change phase already in progress. • RESTART NOT COMPLETE This IMS initialized before the online change was initiated, but has not completed restart. The online prepare or abort phase is rejected as long as this IMS is in this state. Cancel this IMS, then abort the online change before attempting the online change prepare phase again. • SHUTDOWN This IMS shut down normally since the last successful online change. Online change is terminated on this IMS.

Example for TERMINATE OLC Command

TSO SPOC input:

```
TERMINATE OLC
```

TSO SPOC output:

```
MbrName  Member     CC
IMS3     IMS2        0
IMS3     IMS3        0
IMS3     SYS3        0
```

OM API input:

```
CMD (TERMINATE OLC)
```

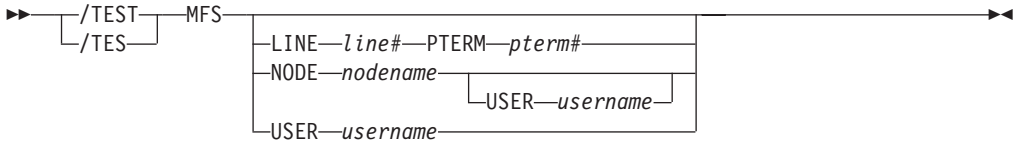
OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2002.163 15:49:27.197919</statime>
<stotime>2002.16315:49:27.712209</stotime>
<staseq>B7C4ADFC0D4DF841</staseq>
<stoseq>B7C4ADFC8ADD1F45</stoseq>
<rqsttkn1>USRT011 10084927</rqsttkn1>
<rc>0200000C</rc>
<rsn>00003000</rsn>
</ctl>
<cmderr>
<mbr name="IMS2    ">
<typ>IMS     </typ>
<styp>DBDC   </styp>
<rc>02000004</rc>
<rsn>00001008</rsn>
</mbr>
<mbr name="SYS3    ">
<typ>IMS     </typ>
<styp>DBDC    </styp>
<rc>02000004</rc>
<rsn>00001008</rsn>
</mbr>
</cmderr>
<cmd>
<master>IMS3    </master>
<userid>USRT011 </userid>
<verb>TERM</verb>
<kwd>OLC  </kwd>
<input>TERMINATE OLC</input>
</cmd>
<cmdrsphdr>
<hdr slbl="MBR"    llbl="MbrName" scope="LCL" sort="n" key="0" scroll="yes" len="8"
dtype="CHAR" align="left" />
<hdr slbl="IMSMBR" llbl="Member"  scope="LCL" sort="a" key="1" scroll="no"  len="8"
dtype="CHAR" align="left" />
<hdr slbl="CC"     llbl="CC"      scope="LCL" sort="n" key="0" scroll="yes" len="4"
dtype="INT"  align="right" />
</cmdrsphdr>
<cmdrspdata>
<rsp>MBR(IMS3    ) IMSMBR(SYS3    ) CC(   0) </rsp>
<rsp>MBR(IMS3    ) IMSMBR(IMS2    ) CC(   0) </rsp>
<rsp>MBR(IMS3    ) IMSMBR(IMS3    ) CC(   0) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: Global online change was aborted for the IMSplex after a successful `INITIATE OLC PHASE(PREPARE)` command. Global online change was successfully terminated.

Chapter 62. /TEST

Format



Environments and Keywords

Table 157 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 157. Valid Environments for the /TEST Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/TEST	X		X
LINE	X		X
NODE	X		X
PTERM	X		X
USER	X		X

Usage

The /TEST command places a terminal or user into either test mode or MFSTEST mode.

In test or echo mode, any input from the terminal is sent back. Input and output errors are not checked, and IMS error notification procedures are bypassed. Echo mode continues until reset with an /END, /IAM, /STOP LINE, /STOP LINE PTERM, or a /STOP NODE command. The /DISPLAY command identifies a terminal or user in test mode or MFSTEST mode. If no keywords are supplied, the terminal entering the command is placed into test mode or MFSTEST mode.

Test mode is not a command significant status, so the commands to set test mode are not recoverable nor are they kept after signons and can only be set by the end user or terminal, not remotely by an operator. The /TEST LINE, /TEST NODE, and /TEST USER commands, which set test mode remotely, are no longer supported.

In MFSTEST mode, terminals supported by message format service use format blocks from a special test library if the requested format block is in the test library; otherwise, the blocks are obtained from the production library. MFSTEST mode continues until reset with an /END command. Certain error conditions can occur that cause MFSTEST mode to terminate. If an error condition occurs, the terminal operator receives an error message.

MFSTEST mode is a command significant status, is recoverable and is remembered across logons and signons. For example, if a /TEST MFS NODE command is entered at a node, the node logs off and logs back on at another

© Copyright IBM Corp. 1974, 2003

terminal, MFSTEST mode is still in effect. If a dynamic user issues a /TEST MFS USER command, signs off, and then signs on again at another terminal, MFSTEST mode is still in effect.

/TEST MFS NODE applies to dynamic nodes in addition to static nodes because MFSTEST mode is associated with dynamic nodes as well as dynamic users. /TEST MFS NODE and /TEST MFS NODE USER set MFSTEST mode at the node level. /TEST MFS USER sets MFSTEST at the user level. /TEST MFS with no keywords sets MFSTEST at the node level for static terminals (they have no user level) and at the user level for dynamic terminals.

The /TEST NODE USER command is supported for static and dynamic ISC sessions. For ISC, /TEST MFS NODE USER is required. You cannot use /TEST MFS NODE (without USER) for ISC and have it apply to all of the half-sessions.

If global resource information is kept in Resource Manager, MFSTEST mode is set globally. If global resource information is not kept in Resource Manager, the resource does not exist, and ETO is enabled, the resource (node or user) is created and MFSTEST mode is set. If a temporary node is dynamically created to hold command status, and the temporary node has MFSTEST status, then, when a logon occurs for the node, the MFSTEST status is set for the logged-on node. If the node logging on is an ISC parallel session, MFSTEST is set only for the first half-session that is logged on. Subsequent ISC half-sessions will not be put into MFSTEST mode.

MFS
Specifies MFS test mode for the terminal or user. The MFS parameter is valid only for terminals supported by Message Format Service (MFS). When the /TEST MFS USER command is issued for a dynamic user, it is not possible to determine if the MFSTEST mode is valid until the user signs on to a terminal. Once the user signs on to a terminal, a check is made to determine whether that terminal supports MFSTEST mode. If the terminal does not support MFSTEST mode, the mode is not propagated to the terminal and is removed from the USER unless another /TEST MFS USER command is issued.

LINE, PTERM
Specifies the BTAM line and pterm to place into MFSTEST mode.

NODE
Specifies the VTAM node to place into MFSTEST mode.

The /TEST MFS NODE form of the command is valid only for statically defined nodes.

The /TEST MFS NODE USER command is valid for ISC, LUP, and 3600. For ISC, the /TEST MFS NODE nodename USER username form of the command is supported for ISC nodes and applies to the half-session allocated to the USER username.

USER
When specified without the NODE keyword, USER specifies the dynamic user to place into MFSTEST mode. MFSTEST mode status is kept from one signon to another. For example, if a user issues a /TEST MFS command, signs off, and then signs on again at another terminal, the MFSTEST mode is still in effect. If the user does not exist, it is created and MFSTEST mode is set.

Examples

Example 1 for /TEST Command

Entry ET:

```
/TEST
```

Response ET:

```
DFS058I  TEST COMMAND COMPLETED
```

Entry ET:

```
NOW IS THE TIME TO COME TO THE AID
```

Response ET:

```
NOW IS THE TIME TO COME TO THE AID
```

Explanation: The entering terminal is placed in echo mode and continues to receive message input as output until test mode is terminated.

Example 2 for /TEST Command

Entry ET:

```
/TEST MFS
```

Response ET:

```
DFS058I  TEST COMMAND COMPLETED
```

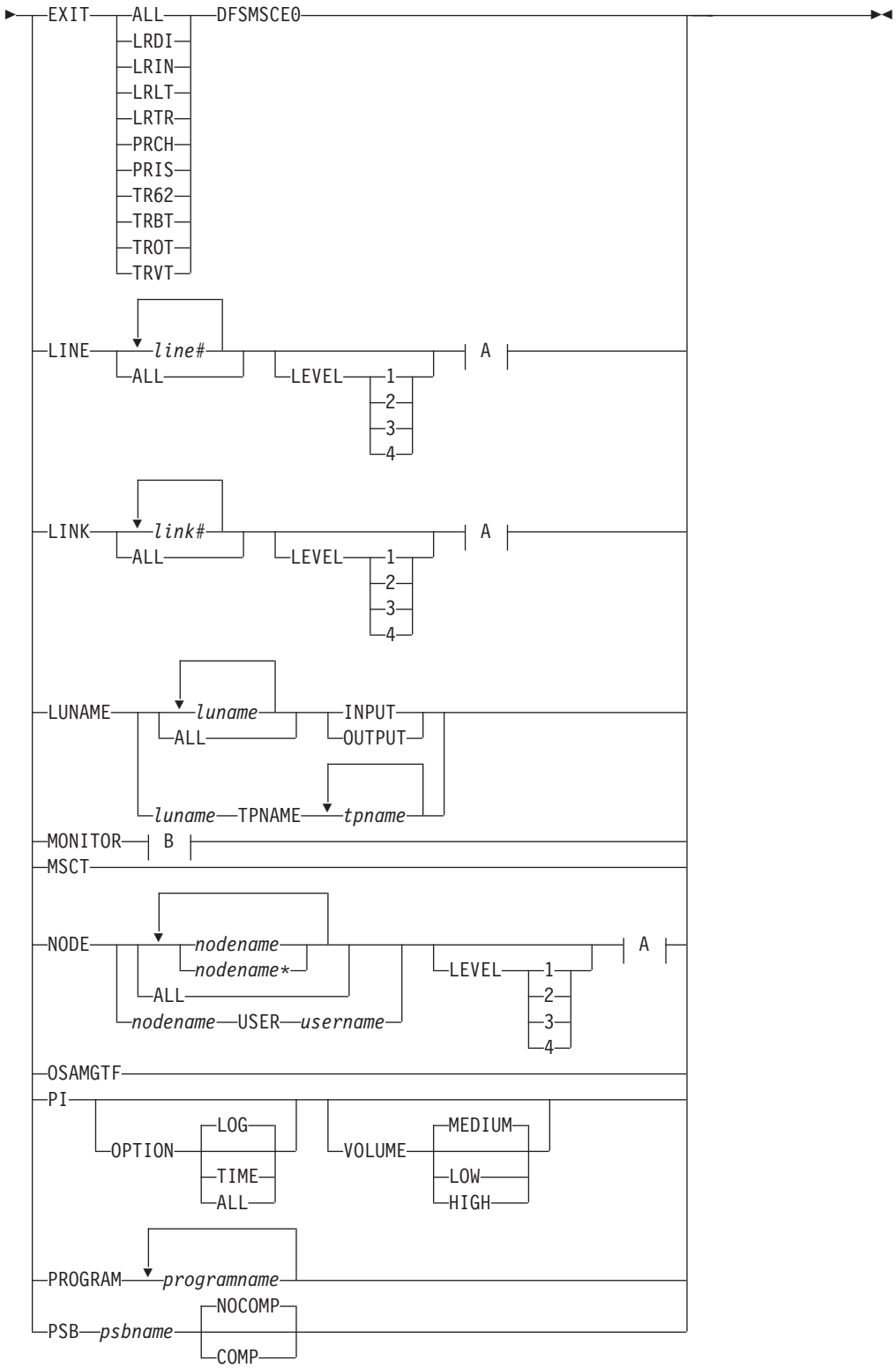
Explanation: The entering terminal is placed into MFSTEST mode.

Chapter 63. /TRACE

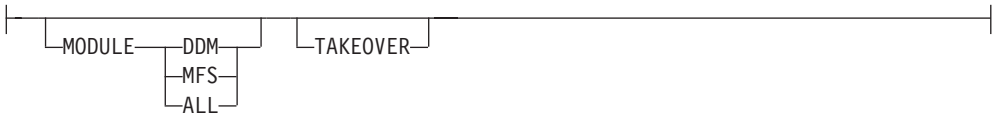
Format

/TRACE Command: EXIT through PSB

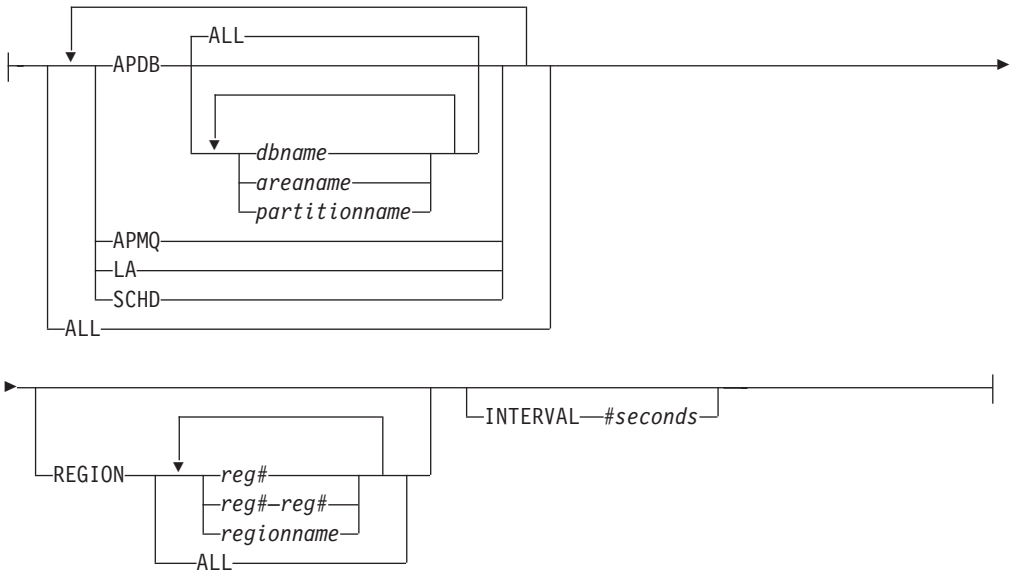
© Copyright IBM Corp. 1974, 2003


EXIT
ALL
LRDI
LRIN
LRLT
LRTR
PRCH
PRIS
TR62
TRBT
TROT
TRVT
DFSMSCE0
LINE
line#
ALL
LEVEL
1
2
3
4
A
LINK
link#
ALL
LEVEL
1
2
3
4
A
LUNAME
luname
ALL
INPUT
OUTPUT
luname
TPNAME
tpname
MONITOR
B
MSCT
NODE
nodename
nodename*
ALL
nodename
USER
username
LEVEL
1
2
3
4
A
OSAMGTF
PI
OPTION
LOG
TIME
ALL
VOLUME
MEDIUM
LOW
HIGH
PROGRAM
programname
PSB
psbname
NOCOMP
COMP

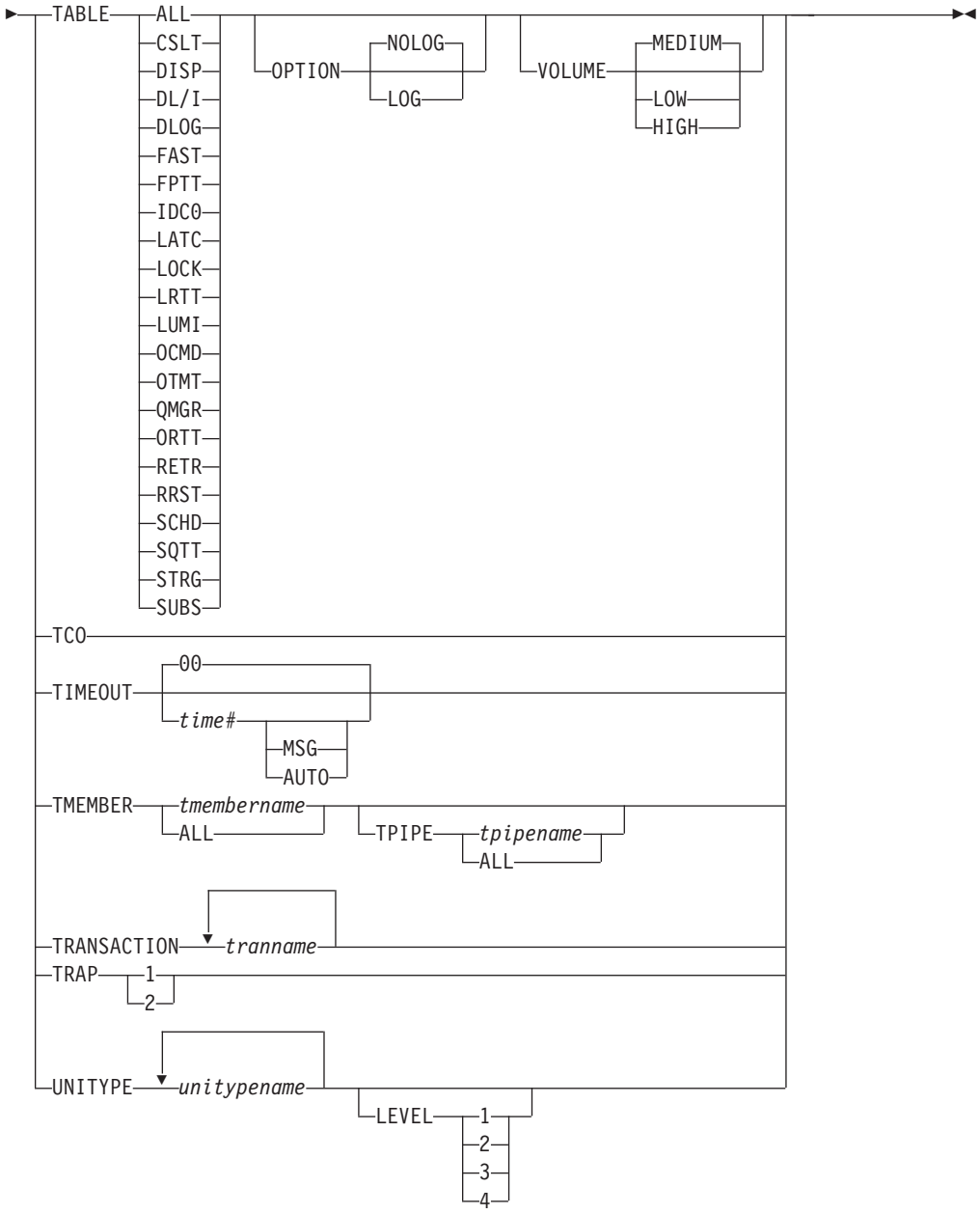
A:



B:



/TRACE Command: TABLE through UNITYPE



Environments and Keywords

Table 158 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 158. Valid Environments for the /TRACE Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/TRACE	X	X	X
AUTO	X		X
COMP	X	X	
EXIT	X		X

Table 158. Valid Environments for the /TRACE Command and Keywords (continued)

Command / Keywords	DB/DC	DBCTL	DCCTL
INPUT	X		X
LEVEL	X		X
LINE	X		X
LINK	X		X
LUNAME	X		X
MODULE	X		X
MONITOR	X	X	X
MSG	X		X
NOCOMP	X	X	
NODE	X		X
OPTION	X	X	X
OSAMGTF	X	X	
OUTPUT	X		X
PI	X	X	
PROGRAM	X	X	X
PSB	X	X	
SET	X	X	X
TABLE	X	X	X
TAKEOVER	X		X
TCO	X	X	X
TIMEOUT	X		X
TMEMBER	X		X
TPIPE	X		X
TPNAME	X		X
TRANSACTION	X		X
TRAP	X		X
UNITYPE	X		X
USER	X		X
VOLUME	X	X	X

Usage

/TRACE directs and controls the IMS capabilities for tracing internal IMS events. It also starts, stops, and defines the activity to be monitored by the IMS Monitor.

The information developed by the LINE, LINK, NODE, UNITYPE, TRANSACTION, PROGRAM, PSB, and TCO keywords is written on the IMS system log (type X'5F' for PSB, and type X'67' records for the other keywords mentioned.) PI (program isolation) and TABLE trace information is kept in storage or logged (type X'67' records), depending on specification of additional keywords. The MONITOR keyword provides no such output on the system log; it only controls the IMS Monitor. The monitor develops its own output data and writes it on a separate data

set. For an explanation of the output reports provided by the monitor, refer to the *IMS Version 9: Utilities Reference: System*.

The status and options of the current IMS traces can be displayed with the /DISPLAY TRACE command.

DFSMSCE0
The TM and MSC Message Routing and Control user exit. When this keyword is used on the /TRACE EXIT command, the traces causes IMS to write type 6701-MSEA and 6701-MSEB log records to the log data set when the exit routine is called. A 6701-MSEA record is logged when the exit is called if the trace is active for the entry point. A 6701-MSEB record is logged when the exit returns to IMS. For more information on the user exit, see the *IMS Version 9: Customization Guide*. For information on printing and interpreting the trace records from the user exit, see the *IMS Version 9: Diagnosis Guide and Reference*.

EXIT
Identifies user exit tracing. When using the EXIT keyword, you must specify one of the following parameters to turn on or off:

ALL The trace is turned on or off for all entry points.

LRDI Calls the Link Receive Direct Routing exit entry point.

LRIN Calls the Link Receive Intermediate exit entry point.

LRLT Calls the Link Receive LTERM exit entry point.

LRTR Calls the Link Receive Transaction exit entry point.

PRCH Calls the Program Routing CHNG Call exit entry point.

PRIS Calls the Program Routing ISRT Call exit entry point.

TR62 Calls the Terminal Routing LU62 exit entry point.

TRBT Calls the Terminal Routing BTAM exit entry point.

TROT Calls the Terminal Routing OTMA exit entry point.

TRVT Calls the Terminal Routing VTAM exit entry point.

LEVEL
Expands the LINE, LINK, NODE, or UNITYPE trace functions. The LEVEL specification is for the entire IMS system and is changed only by reissuing /TRACE with different values or by restarting the IMS control region.

LEVEL indicates the extent of the control block trace information desired. The indicated control blocks are only traced at relevant times. All levels are inclusive of numerically lower levels. The following list displays the levels and their associated blocks.

Level	Blocks
1	CLB (DECB) or LLB(MSC)
	CTB or LTB(MSC)
	IOB (for BTAM lines) or IOSB (MSC for channel-to-channel links)
2	CNT or LNB(MSC)
	CXB
	CRB

CIB

CCB

PD stack

3 queue manager buffers

Input/output line buffers

LXB (for channel-to-channel links and processor storage to processor storage)

4 save area sets (IMS dispatching)

If the first /TRACE SET ON command does not specify LEVEL, a default of 4 will be used. Specifying LEVEL on subsequent commands will change the defaults.

LINE
Enter this keyword (and the associated parameters) to cause events related to the lines to be traced.

LINK
Enter this keyword (and the associated parameters) to cause events related to the logical links to be traced.

LUNAME TPNAME
Activates and deactivates tracing for a particular LU name or TP name of the LU name. Specifying this command causes trace entries to be written to the LUMI trace table. For this reason, the /TRACE SET ON TABLE LUMI command must be entered first in order to create the table for trace entries that will be created by subsequent /TRACE SET ON LUNAME commands. A trace entry is written:

- On LU 6.2 module entries/exits
- When APPC calls are made
- When errors are encountered

The INPUT and OUTPUT keywords provide the operator with the flexibility to control the volume of trace data for LU 6.2 devices.

Specifying the keyword INPUT with the LUNAME keyword indicates tracing is activated or deactivated for input and synchronous outbound activities. Specifying the parameter ALL with the INPUT keyword causes all future LU 6.2 input and synchronous outbound activities to be traced as well.

Specifying the keyword OUTPUT with the LUNAME keyword indicates tracing is activated or deactivated for asynchronous outbound activities. Specifying the parameter ALL with OUTPUT causes all future LU 6.2 asynchronous outbound activities to be traced as well.

Specifying neither INPUT or OUTPUT is the same as both INPUT and Tracing is activated or deactivated for input and both synchronous and asynchronous outbound activities. Specifying the parameter ALL in this case causes all future LU 6.2 inbound activities, synchronous and asynchronous outbound activities to be traced as well.

The network-qualified LU name is optional for the LUNAME keyword. If the LU name is not a network-qualified LU name and no TP name is specified, tracing is activated or deactivated for all the network-qualified LU names in the system whose LU name matches the LU name specified.

If the specified resource does not exist and tracing is activated, a structure is created to retain the status.

MODULE
Is used to expand the LINE, LINK, NODE, or UNITYPE trace functions. The MODULE specification is for the entire IMS system and is changed only by reissuing /TRACE with different values or by restarting the IMS control region.

MODULE indicates which modules are to have their control blocks traced.

ALL Both DDM and MFS

DDM Communication analyzer and device-dependent module interfaces

MFS Communication analyzer and message format service module interfaces

If the first /TRACE SET ON command does not specify MODULE, a default of ALL will be used. Specifying MODULE on subsequent commands will change the defaults.

MONITOR
Enter this keyword to activate or deactivate the IMS Monitor.

When activating the monitor, you must specify one or more of the following MONITOR parameters to indicate the events to be monitored:

ALL Monitor all of the activity in this list.

APDB Monitor activity between application programs and databases, including Fast Path activity. Monitoring includes all application program requests to external subsystem databases. Monitoring can optionally be limited to a subset of full-function databases or partitions, Fast Path DEDBs and MSDBs, and the areas comprising those DEDBs by specifying database names or area names.

You can specify the ALL parameter to indicate all databases, areas, and partitions, or you can explicitly enter database names, area names, and partition names.

APMQ Monitor activity between application programs and message queues, including Fast Path activity.

INTERVAL Monitor events for a fixed interval of time, entered in seconds. INTERVAL defines the period of time **after which** no monitor log records will be written.

The duration of the monitoring must be less than twenty-four hours (86,400 seconds). When INTERVAL is not specified, monitoring will continue until the /TRACE SET MONITOR OFF command is issued, or until IMS shuts down.

INTERVAL does not define when the Monitor will be turned off, because the IMS Monitor will not be turned off until the first attempt is made to write a monitor log record after the defined interval has expired.

LA Monitor line and logical link events.

REGION Monitor events related to specific dependent regions. The regions might or might not currently be active. Each region can be specified as:

- A region number from 1 to 999 (*reg#*). The number cannot exceed the MAXPST with which IMS was brought up.

- A range of region numbers from 1 to 999 (*reg#–reg#*). The number cannot exceed the MAXPST with which IMS was brought up.
- A region name (*regionname*).

If REGION is not specified, or when REGION ALL is specified, the activities of all dependent regions are monitored.

SCHD Monitor scheduling and termination events, including Fast Path activities.

You can specify any combination of `ALL`, `APDB`, `APMQ`, `INTERVAL`, `LA`, `REGION`, and `SCHD` parameters on the MONITOR keyword, as shown in Table 159

Table 159 lists the environments (DB/DC, DBCTL, and DCCTL) from which the MONITOR keyword parameters can be issued.

Table 159. MONITOR Keyword Parameter Environments

Keyword Parameter	DB/DC	DBCTL	DCCTL
ALL	X	X	X
APDB	X	X	
APMQ	X		X
INTERVAL	X	X	X
LA	X		X
REGION	X	X	X
SCHD	X	X	X

The IMS Monitor report output varies depending upon which keywords or parameters you specify. Sections of the report can be misleading if required records are excluded. For example, if you specify the APDB parameter without the SCHD parameter, PSB/PCB relationships will not be correctly represented. To get the total DL/I call reports without the IMS line activity, the correct parameters to specify are APDB, APMQ, and SCHD.

The monitor writes log records until one of the following occurs:

- `/TRACE SET OFF MONITOR` is entered.
- The time interval specified by the INTERVAL parameter is reached.
- IMS is shut down.

When deactivating the monitor, no parameters are required. Any parameters that are entered, other than ALL, are ignored. ALL is the default. If the monitor is to be reactivated, a new set of MONITOR keyword parameters must be selected.

NODE

Causes events related to the node or nodes to be traced.

The NODE parameter can be generic if the USER keyword is not specified and applies to nodes that already exist. Generic NODE parameters do not cause any dynamic nodes to be created.

For ISC nodes, the /TRACE NODE nodename without the USER applies to all half-sessions for NODE nodename, including dynamic ISC sessions that are dynamically allocated later.

For nodes that do not exist, /TRACE SET ON NODE nodename without the USER keyword causes the dynamic NODE nodename to be created to maintain knowledge of the trace request when the node becomes active. Until the node becomes active, /DISPLAY NODE shows a type of UNK (unknown). If the trace is subsequently turned off and the temporary node still exists, it is deleted at the next checkpoint.

/TRACE NODE nodename USER username is valid:

- For ISC nodes and dynamic nodes with signed on users
- Only if the USER username is still allocated to NODE nodename
- For non-ISC nodes only if the USER username is still signed on to NODE nodename

If global resource information is kept in Resource Manager, /TRACE NODE sets a global trace status for the node and sets the trace status locally. If global resource information is not kept in Resource Manager, /TRACE NODE sets the trace status locally. If the node does not exist in an ETO environment, IMS creates the node and sets trace status for the local node.

OSAMGTF
Causes the OSAMGTF keyword to activate or deactivate the OSAM GTF (Generalized Trace facility).

PI Causes program isolation trace entries to be written to a trace table. PI trace entries are written in the same trace table as DL/I and lock activity trace entries. A PI trace entry contains information about program isolation ENQ/DEQ calls and DL/I calls. The trace entry created by /TRACE TABLE DLI contains different information about DL/I calls and is written as a separate entry in the same trace table. Starting the LOCK trace also causes PI tracing to occur.

If PI is entered without the OPTION keyword, the program isolation trace is kept in storage without being logged. If you are using the program isolation trace to provide statistics and performance data, you should enter OPTION(ALL).

Table 160 lists various /TRACE command formats and shows whether the command influences tracing, logging, and the additional time field.

Table 160. /TRACE Command Formats

Command	Tracing	Logging	Additional Time Field
/TRACE SET ON PI	Yes	No	No
/TRACE SET OFF PI	No	No	No
/TRACE SET ON PI OPTION[1]	Yes	Yes	No
/TRACE SET OFF PI OPTION[1]	Yes	No	No
/TRACE SET ON PI OPTION TIME	Yes	No	Yes
/TRACE SET OFF PI OPTION TIME	Yes	No	No
/TRACE SET ON PI OPTION ALL	Yes	Yes	Yes
/TRACE SET OFF PI OPTION ALL	Yes	No	No

Note:

1. This is the same command as /TRACE SET ON/OFF PI OPTION LOG.

OPTION
Indicates a request for one of the following program isolation trace options:

ALL Both LOG and TIME.

LOG Requests that traced data be written to the system log. If SET OFF, tracing continues but no buffers are transferred to the system log. LOG is the default.

The log option includes the possibility of externally tracing to a data set other than the IMS OLDS. If specified, DASD external tracing has first priority, TAPE external tracing has second priority, and IMS OLDS has third priority. External tracing to the OLDS is not done without operator approval. EXTERNAL trace is available to the alternate system only for DASD/TAPE type, but not for OLDS. For more information, see *IMS Version 9: Diagnosis Guide and Reference* and *IMS Version 9: Operations Guide*.

TIME Requests that an additional time field be included in each ENQ/DEQ request trace record if a WAIT was needed. This field will contain elapsed wait time. If set OFF, tracing continues but only the time of day is recorded.

Entries for Fast Path have no elapsed wait time.

VOLUME
Specifies the volume of entries to be written to the PI trace table: LOW volume, MEDIUM volume (default), or HIGH volume.

PROGRAM
Is used to trace the DL/I portion of Data Communications (DC) for a specific program. Each DL/I call to a TPPCB, issued by the user application program, is traced on entry to and exit from the DC call handler DFSDLA30. On entry to DFSDLA30 a type 6701-LA3A record is written, on exit from DFSDLA30 a type 6701-LA3B record is written.

Each record will contain the following items if applicable:

- TPPCB
- Up to 64 bytes of the I/O area
- SMB
- PST

If the batch message program (BMP) being traced is the Queue Control Facility (QCF) program product 5697-E99, a 6701-MRQB record is logged by the QCF module DFSQMRQ0. The default program name for the QCF BMP is MRQPSB, and can be overridden on the MSGQUEUE SYSGEN macro.

Items logged in the 6701-MRQB record, if applicable, are:

- TPPCB
- AIB
- I/O AREA
- PST
- QTPDST
- QSAPWKAD
- QMBA
- PSTDCA
- REG14-12

For more information about the QCF module DFSQMRQ0 and the 6701-MRQB trace, refer to *IMS Version 9: Diagnosis Guide and Reference*. For more information about the QCF product, refer to *IMS Queue Control Facility for z/OS, User's Guide V1R2 (SC26-9685-02)*.

When CPI Communications driven transaction programs issue the DL/I APSB call specifying a PSB that contains alternate PCBs, only the PROGRAM keyword is applicable.

PSB

Records all full function IMS DL/I database calls issued for the named PSB. FP/DC/SAA calls are not captured when `/TRACE SET ON PSB` initiated tracing of PSBs.

For LU 6.2, the PSB keyword is applicable only if the CPI Communications driven transaction program has issued a DL/I APSB call to allocate a PSB.

COMP Used with the `/TRACE SET PSB` command to generate PCB and data-compare statement images.

The `/TRACE SET PSB psbname COMP` command only applies to BMPs in a DBCTL environment.

NOCOMP Prevents PCB and data-compare statement images from being generated. NOCOMP is the default.

The information resulting from the use of this keyword is written on the X'5F' log record.

SET

This keyword turns ON or OFF one of the following:

- The control block trace for particular LINE, LINK, NODE, or UNITYPE
- The program isolation trace
- The IMS Monitor
- The DL/I call trace for a specific PSB
- Tracing into a specific trace table
- Tracing of the DL/I portion of DC for programs and transactions
- Traps that enable MFS serviceability aids
- The VTAM I/O Detection facility
- The TCO trace
- The OSAMGTF trace
- Tracing for a particular LUNAME

TABLE

Is used with the SET keyword to start or stop online tracing into the specified trace tables. The TABLE keyword parameter indicates the specific trace that is to be activated or deactivated. The DL/I, LOCK, and PI traces share the same trace tables. However, turning on the DL/I trace does not turn on the LOCK trace, and vice versa.

The following trace into trace tables can be turned on or off with the online `/TRACE` command.

Table 161 on page 689 shows the environments in which the trace tables are valid.

Table 161. Trace Tables and Environments in Which They Are Valid

Trace Table	DB/DC	DBCTL	DCCTL
ALL	X	X	X
CSLT	X	X	X
DISP	X	X	X
DL/I	X	X	X
DLOG	X	X	X
FAST	X	X	X
FPTT	X	X	X
IDC0	X		X
LATC	X	X	X
LOCK	X	X	
LRTT	X	X	X
LUMI	X		X
MSCT	X		X
OCMD	X	X	X
ORTT	X		X
OTMT	X	X	
QMGR	X		X
RETR	X	X	
RRST	X		X
SCHD	X	X	X
SQTT	X		X
STRG	X	X	X
SUBS	X	X	X

ALL Indicates that traces into all trace tables are to be enabled or disabled. This is the default.

CSLT Indicates that the CSL trace is to be activated or deactivated.

DISP Indicates that the dispatcher trace is to be activated or deactivated.

DL/I Indicates that DL/I tracing is to be activated or deactivated.

DLOG Indicates that the logging trace is to be activated or deactivated.

FAST Indicates that the Fast Path trace is to be activated or deactivated.

Fast Path Trace is activated by including the FPTRACE DD statement in the dependent region JCL to define the destination of the trace output and by issuing the operator command `/TRACE SET ON TABLE FAST`.

Recommendation: Only run this trace in a test environment because the FPTRACE output is very large.

FPTT Indicates that the Fast Path table trace is to be activated or deactivated.

IDC0 Indicates that tracing of errors in modules DFSCNXA0 and DFSIDC00 is to be activated or deactivated.

LATC Indicates that the latch trace is to be activated or deactivated.

LOCK Indicates that LOCK and PI tracing is to be activated or deactivated.

LRTT Indicates that the log router trace table is to be activated or deactivated. `LRTT` is only valid for an RSR tracking subsystem.

LUMI Indicates that the LU 6.2 manager trace is to be activated or deactivated. `/TRACE SET ON TABLE LUMI` must be entered first before entering any `/TRACE SET ON LUNAME` command in order to create the LUMI trace table for trace entries.

LUMI is not valid for an RSR tracking subsystem.

2 **MSCT** Indicates that the MSC trace is to be activated or deactivated.

OCMD Indicates that the OM command trace is to be activated or deactivated.

ORTT Indicates that the Online Recovery System trace is to be activated or deactivated.

OTMT Indicates that the IMS Open Transaction Manager Access (OTMA) trace is to be activated or deactivated.

QMGR Indicates that the queue manager trace is to be activated or deactivated.

RETR Indicates that the DL/I retrieve trace is to be activated or deactivated. RETR is not valid for an RSR tracking subsystem.

RRST Indicates that the Resource Recovery trace is activated or deactivated.

SCHD Indicates that the scheduler trace is to be activated or deactivated. SCHD is not valid for an RSR tracking subsystem.

SQTT Indicates that the shared queues trace is to be activated or deactivated. SQTT is only valid in a shared-queues environment.

STRG Indicates that the storage manager trace is to be activated or deactivated.

SUBS Indicates that the external subsystem trace is to be activated or deactivated. SUBS is not valid for an RSR tracking subsystem.

OPTION

Is used to indicate whether or not the trace tables are to be logged.

The meaning of the OPTION parameters are:

LOG Requests that the trace tables be written to the system log before they are reused (wrapped).

The log option includes the possibility of externally tracing to a data set other than the IMS OLDS. If specified, DASD external tracing has first priority, TAPE external tracing has second priority, and IMS OLDS has third priority. External tracing to the OLDS is not done without operator approval. This option is ignored for the retrieve trace and the Fast Path trace, (default).

FPTRACE has its own SYSOUT. EXTERNAL trace is available to the alternate system only for DASD/TAPE type, but not for OLDS. For more information, see *IMS Version 9: Diagnosis Guide and Reference* and *IMS Version 9: Operations Guide*.

NOLOG Indicates that the trace tables are not logged. This means that there will be no record of trace tables that have been reused. NOLOG is the default.

The same trace table is used for DL/I, LOCK, and PI trace information. Whether or not this table is logged will depend on the logging option of the last /TRACE command that was entered for DL/I, PI, or LOCK.

VOLUME

Specifies the volume of entries to be written to the specified trace table: LOW volume, MEDIUM volume (default), or HIGH volume.

The VOLUME keyword is valid on active and RSR tracking subsystems.

TAKEOVER

Controls tracing during takeover only, and is separate from regular tracing. TAKEOVER tracing can be set for LINE, LINK, NODE, and UNITYPE keywords.

TAKEOVER only applies in an XRF environment. When TAKEOVER is used with SET OFF, the trace is turned off before takeover. When an output message is dequeued for a terminal, takeover tracing will stop for that terminal.

If both regular and takeover tracing are entered, the most recent setting will override any previous settings; for example, takeover tracing will override regular tracing if regular tracing was entered first. This means that either regular or TAKEOVER tracing can be in effect, but not both.

/TRACE TAKEOVER can be issued only from an XRF active system. It is rejected if entered from an alternate or non-XRF system. It is recovered across restart and takeover, and only needs to be entered once until cold start. Tracing occurs only if the session was active at the time of the takeover.

TCO

Is used to trace TCO (Time Controlled Operation) activity. For the first /TRACE command with the TCO keyword, the default module and level information is used. TCO trace is basically a DC LINE or NODE trace, and the information developed is also written on the type X'67' log record.

TIMEOUT

Is used to start or stop the I/O Timeout Detection facility.

time#

is the number of minutes used to determine if the I/O response is overdue. After this number of minutes, time has run out for the response. The range is from 1 through 60; the default value is 0.

If time# is 0, or not specified, then when I/O is initiated for a node, the node will be placed on a queue, so that its status can be displayed with the /DISPLAY TIMEOVER command. You will not be notified if the node does not receive a response and the time elapses, and the node will not be reactivated.

If time# is not 0, then the following keywords can be used.

MSG
Indicates that a message is issued to the master terminal when I/O takes longer than time# minutes. The message indicates that the time has elapsed.

AUTO
IMS issues a message to the master terminal, then perform a VTAM VARY NET,INACT and a VARY NET,ACT, if I/O takes longer than time# minutes. An /OPNDST is performed for operable devices that are not shared. For ISC nodes, a message is issued, but there is no automatic (AUTO) restart of any sessions and no VTAM VARY commands issued.

If you wish to change the time period or the action to be taken if timeout occurs, you can enter the /TRACE SET ... TIMEOUT command while the Timeout Detection facility is already active. However, if nodes are receiving or sending input or output, they will function according to the previous settings of the /TRACE ... TIMEOUT command. If this is undesirable, then you should enter the /TRACE SET OFF TIMEOUT command before reentering /TRACE SET ... TIMEOUT.

If the timeout trace facility failed during IMS initialization, the /TRACE SET ... TIMEOUT command is rejected with an error message.

The VTAM TIMEOUT I/O facility is automatically started during IMS shutdown. It is set for 1 minute and AUTO.

TMEMBER
Is used to trace IMS Open Transaction Manager Access (OTMA) client activity for OTMA clients.

TPIPE
Is used to trace transaction pipe activity for OTMA clients.

While processing the /TRACE TMEMBER TPIPE command, IMS creates a temporary transaction pipe (if one does not already exist) with the trace status. IMS sets the synchronization status for this transaction pipe when it sends or receives the first message for the transaction pipe.

TRANSACTION
Is used to trace the DL/I portion of Data Communications (DC) for a specific transaction. Each DL/I call to a TPPCB, issued by the user application program, is traced on entry to and exit from the DC call handler DFSDLA30. Upon entry to DFSDLA30, a type 6701-LA3A record is written; upon exit from DFSDLA30, a type 6701-LA3B record is written.

Each record will contain the following items, if applicable:

- TPPCB
- Up to 64 bytes of the I/O area
- SMB
- PST

For more information about this trace, see the discussion of the DLA3LOG trace in *IMS Version 9: Diagnosis Guide and Reference*.

TRAP
Is used to detect overwrites of MFS blocks. When /TRACE is used, IMS attempts to detect overwrites in the MFS blocks. If an overwrite occurs, IMS sends a warning message. See *IMS Version 9: Diagnosis Guide and Reference* for a description of trace records.

UNITYPE

Causes events related to the physical terminals of specified type to be traced.

The UNITYPE keyword is used to trace all terminals of a specific type. Parameters (unitypename) are similar to the identifiers displayed in the TYPE column by the /DISPLAY NODE and /DISPLAY LINE/PTERM commands. Table 162 shows the terminal types for UNITYPE parameters.

Table 162. UNITYPE Parameters and Terminal Types

UNITYPE Parameter	Terminal Type
2260R	2260/2265 REMOTE
3286	3284/3286
SYSTEM/7	SYSTEM 7
2980	2980
3270R	3270 REMOTE
3270I	3270 LOCAL
RDR/PTR	LOCAL SYSIN/SYSOUT
SYSTEM/3	SYSTEM 3
FIN	3600
3277	3270 VTAM
SYS/7BS	SYSTEM 7 BSC
SLU1	SLU TYPE 1
SLU2	SLU TYPE 2
SLUP	SLU TYPE P
LU 6	LU TYPE 6
NTO	NTO
CONSOLE	z/OS SYSTEM CONSOLE
TWX	TWX SWITCHED
3275SW	3270 SWITCHED
MSCBSC	Binary Synchronous Communication
MSCMTM	Memory to Memory Communication
MSCCTC	Channel to Channel Adapter
MSCVTAM	MSC VTAM

If global resource information is kept in Resource Manager, /TRACE UNITYPE sets a global trace status for all of the nodes of a specific type. This requires that the inactive static nodes be processed on every IMS system. The UNITYPE keyword is similar to specifying a generic parameter. If global resource information is not kept in Resource Manager, /TRACE UNITYPE sets the trace status locally.

Examples

Example 1 for /TRACE Command

To turn on message format service module and communication analyzer level 4 control block tracing for all physical terminals on line 4.

Entry ET:

```
/TRACE SET ON LINE 4 LEVEL 4 MODULE MFS
```

Response ET:

```
DFS058I  TRACE COMMAND COMPLETED
```

Example 2 for /TRACE Command

To turn on tracing for a logical link:

Entry ET:

```
/TRACE SET ON LINK 2 LEVEL 4 MODULE ALL
```

Response ET:

```
DFS058I  TRACE COMMAND COMPLETED
```

Example 3 for /TRACE Command

Entry ET:

```
/TRACE SET ON MONITOR ALL
```

Response ET:

```
DFS058I TRACE COMMAND COMPLETED
DFS2500I DATASET DFSDCMON SUCCESSFULLY ALLOCATED
DFS2212I DC MONITOR STARTED
```

Explanation: The monitor is activated and all events will be monitored.

Entry ET:

```
/TRACE SET OFF MONITOR
```

Response ET:

```
DFS058I TRACE COMMAND COMPLETED
DFS2500I DATASET DFSDCMON SUCCESSFULY DEALLOCATED
DFS2212I DC MONITOR STOPPED
```

Explanation: The monitor is deactivated.

Entry ET:

```
/TRACE SET ON MONITOR LA
```

Response ET:

```
DFS058I TRACE COMMAND COMPLETED
```

Explanation: The monitor is activated. Line and logical link activities will be monitored. The monitoring of events from the previous activation of the monitor no longer apply.

Example 4 for /TRACE Command

To turn on program isolation tracing, include the additional time field in the trace record and have the trace information logged:

Entry ET:

```
/TRACE SET PI OPTION ALL
```

Response ET:

```
DFS058I  TRACE COMMAND COMPLETED
```

Example 5 for /TRACE Command

To turn off logging of program isolation trace data but continue the trace in storage:

Entry ET:

```
/TRACE SET OFF PI OPTION LOG
```

Response ET:

```
DFS058I  TRACE COMMAND COMPLETED
```

Example 6 for /TRACE Command

To stop program isolation tracing.

Entry ET:

```
/TRACE SET OFF PI
```

Response ET:

```
DFS058I  TRACE COMMAND COMPLETED
```

Example 7 for /TRACE Command

To trace all DL/I calls issued for PSB AALST:

Entry ET:

```
/TRACE SET ON PSB AALST COMP
```

Response ET:

```
DFS058I  TRACE COMMAND COMPLETED
```

Example 8 for /TRACE Command

To turn on online tracing into the DL/I trace table:

Entry ET:

```
/TRACE SET ON TABLE DL/I
```

Response ET:

```
DFS058I TRACE COMMAND COMPLETED
```

Example 9 for /TRACE Command

To turn on the dispatcher's trace tables and have them written to the system log:

Entry ET:

```
/TRACE SET ON TABLE DISP OPTION LOG
```

When the dispatcher's trace tables are no longer required:

Entry ET:

```
/TRACE SET OFF TABLE DISP
```

Response ET:

```
DFS058I  TRACE COMMAND COMPLETED
```

Example 10 for /TRACE Command

To turn on storage manager trace tables:

Entry ET:

```
/TRACE SET ON TABLE STRG
```

Response ET:

```
DFS058I TRACE COMMAND COMPLETED
```

When the storage manager trace tables are no longer needed:

Entry ET:

```
/TRACE SET OFF TABLE STRG
```

Response ET:

```
DFS058I TRACE COMMAND COMPLETED
```

Example 11 for /TRACE Command

Entry ET:

```
/TRACE SET ON TMEMBER CLIENT1 TPIPE TPIPESY
```

Response ET:

```
DFS058I 15:45:05 TRACE COMMAND COMPLETED   SYS3
DFS996I *IMS READY*  SYS3
```

Example 12 for /TRACE Command

To log the PCB, I/O area, and PST whenever module DFSDLA30 is invoked to process transaction APPLE:

Entry ET:

```
/TRACE SET ON TRANSACTION APPLE
```

Response ET:

```
DFS058I TRACE COMMAND COMPLETED
```

Example 13 for /TRACE Command

To trace all 2740 model 2 terminals:

Entry ET:

```
/TRACE SET ON UNITYPE 2740II
```

Response ET:

```
DFS058I TRACE COMMAND COMPLETED
```

Note: The control blocks and modules traced depend on the last LEVEL and MODULE specifications entered before the above command. If the LEVEL and MODULE keywords have not yet been entered, the defaults of 4 and ALL will be used.

Example 14 for /TRACE Command

This is an example of temporary nodes created to retain trace status data.

Entry ET:

```
/TRACE SET ON NODE DTSLU607
```

Response ET:

```
DFS058 TRACE COMMAND COMPLETED
```

Entry ET:

```
/DISPLAY NODE DTSLU607
```

Response ET:

```
NODE-USR TYPE   CID       RECD ENQCT DEQCT   QCT  SENT
DTSLU607 UNK    00000000     0     0     0     0     0 TRA
*90127/091634*
```

Entry ET:

```
/DISPLAY NODE DTSLU607
```

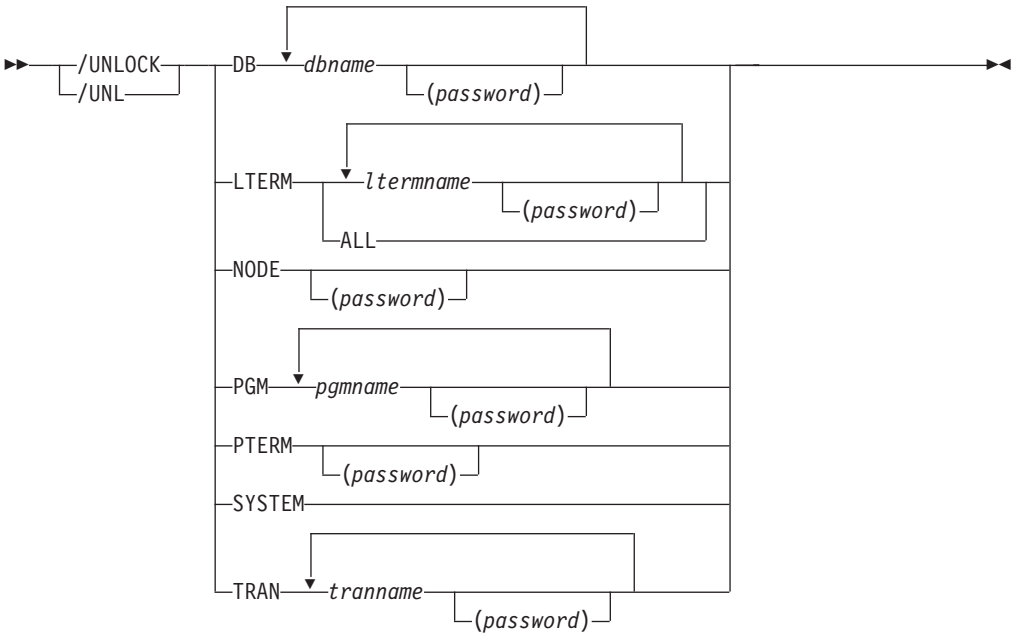
Response ET:

```
NODE-USR TYPE   CID       RECD ENQCT DEQCT   QCT  SENT
DTSLU607 LUT6
-N/A     UNK    00000000     0     0     0     0     0 TRA
-IMSUS01        01000002     0     0     0     0     0 SIGN(IMSUS01 )
                                                       IDLE CON TRA PRI
-IMSUS02        01000004     0     0     0     0     0 SIGN(IMSUS02 )
                                                       IDLE CON TRA PRI
*90127/091432*
```

Explanation: The /TRACE command is issued for a dynamic ISC NODE that does not yet exist, DTSLU607, causing a temporary node to be created to retain the trace status. Once the ISC parallel sessions IMSUS01 and IMSUS02 are allocated, the trace status is applied to them both.

Chapter 64. /UNLOCK

Format



Environments and Keywords

Table 163 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keywords can be issued.

Table 163. Valid Environments for the /UNLOCK Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
/UNLOCK	X	X	X
DB	X	X	
LTERM	X		X
NODE	X		X
PGM	X	X	X
PTERM	X		X
SYSTEM	X		X
TRAN	X		X

Usage

/UNLOCK releases resources that, in most cases, have been previously locked by a /LOCK command.

The /UNLOCK command can be used on HALDBs. For more information see Appendix H, “High Availability Large Database Commands,” on page 867

© Copyright IBM Corp. 1974, 2003

If the terminals are on a switched communication network and a physical or logical terminal disconnection occurs, an implied /UNLOCK is processed against the physical terminal and inquiry logical terminal.

When using ISC, the /UNLOCK command can only be used with logical terminals assigned to allocated users.

/UNLOCK LTERM, NODE, and PTERM apply only to the entering physical terminal.

The /UNLOCK command with the LTERM, NODE, or PTERM keyword is not allowed from the OM API.

/UNLOCK DATABASE, PROGRAM, and TRANSACTION are only valid if entered from the master terminal, the system console, a TCO script, or from an AOI application program. /UNLOCK SYSTEM is only valid if it is entered from the master terminal or from the system console on an XRF system.

3 You can specify password security on the /UNLOCK command by requiring a
3 password after a resource name is entered. If the resource is not defined with
3 password protection in SMU or SMU security is not used, the password is ignored.

3 A resource name can also be defined with password protection in SAF. If the
3 parameter, LOCKSEC=Y (N is the default) is specified on the DFSDCxxx
3 IMS.PROCLIB member, the SAF and user exit calls are made after the SMU calls.
3 If the resource is not defined to SAF, or is defined and is authorized to the user, the
3 command is processed. If the resource is defined to SAF but not authorized for use,
3 the command is rejected with a DFS3689W message.

DATABASE
Specifies the database to be unlocked.

For the results of issuing this command on a shared secondary index, see Appendix D, “Shared Secondary Index Database Commands,” on page 849

The output of the /UNLOCK DB command is changed when the command is entered through the OM API. In this case, the DFS058I message is not returned to OM. The command response returned to OM contains one or more of the following messages: DFS0488I, DFS3466I, DFS132

LTERM
Specifies the logical terminal to be unlocked. This keyword applies only to the entering physical terminal and to logical terminals assigned to that physical terminal.

2 Any password specified with the /UNLOCK LTERM command is ignored for
2 ACF/VTAM nodes created dynamically. Password support on the /UNLOCK
2 command uses the Security Maintenance Utility, which is not supported for
2 dynamic terminals. The password is also ignored for static LTERMs when SMU
2 security is not used, or the LTERM is not defined with password security.

The /UNLOCK LTERM ALL command can only be used when all of the logical terminals associated with the entering physical terminal do not have passwords.

NODE
Specifies the VTAM node to be unlocked. This keyword applies only to the entering physical terminal and to logical terminals assigned to that physical terminal.

2 Any password specified with the /UNLOCK NODE command is ignored for
2 ACF/VTAM nodes created dynamically. Password support on the /UNLOCK

command uses the Security Maintenance Utility, which is not supported for dynamic terminals. The password is also ignored for static nodes when SMU security is not used, or the node is not defined with password security.

PROGRAM
Specifies the application program to be unlocked.

PTERM
Specifies the physical terminal to be unlocked. This keyword applies only to the entering physical terminal and to logical terminals assigned to that physical terminal.

SYSTEM
Notifies a newly created active system in an XRF complex that I/O prevention is complete. I/O prevention is initiated at takeover to ensure that the failing active system cannot write to the databases. The alternate system then initiates I/O toleration to ensure database integrity and to enable new transaction processing as soon as possible. `/UNLOCK SYSTEM` ends the I/O toleration phase of processing. See the *IMS Version 9: Operations Guide* for a more detailed description of I/O prevention, I/O toleration, and the /UNLOCK SYSTEM command.

TRANSACTION
Specifies the transaction code to be unlocked.

`/UNLOCK TRANSACTION` cannot be used for Fast Path exclusive or CPI Communications driven transaction programs.

Examples

Example 1 for /UNLOCK Command

Entry ET:

```
/UNLOCK DATABASE TREEFARM
```

Response ET:

```
DFS058I  UNLOCK COMMAND COMPLETED
```

Explanation: Database TREEFARM is unlocked and can be used.

Example 2 for /UNLOCK Command

Entry ET:

```
/UNLOCK PROGRAM APPLETRE
```

Response ET:

```
DFS058I  UNLOCK COMMAND COMPLETED
```

Explanation: Application program APPLETRE is unlocked and can be executed.

Example 3 for /UNLOCK Command

Entry ET:

```
/UNLOCK PTERM
```

Response ET:

```
DFS058I  UNLOCK COMMAND COMPLETED
```

Explanation: The physical terminal from which the command is entered is unlocked.

Example 4 for /UNLOCK Command

Entry ET:

```
/UNLOCK TRANSACTION SEED
```

Response ET:

```
DFS058I  UNLOCK COMMAND COMPLETED
```

Explanation: Transaction SEED is unlocked and can be scheduled.

Example 5 for /UNLOCK Command

The following three figures illustrate the use of the /UNLOCK SYSTEM command on a newly created active system. Each figure is a formatted master screen for the newly created active system IMSB.

Figure 8 shows a screen of a newly created active system in the I/O toleration phase of processing (awaiting I/O prevention). Database DD41M803 has an I/O toleration EEQE.

```
02/05/15 16:19:03  RSENAME: DFSRSENM  ACTIVE  AWAITING I/O PREVENTION  IMSB
DFS2716I NO MSDBS FOUND - NO MSDB CHECKPOINT TAKEN
 DFS994I *CHKPT 85135/152931**SIMPLE**
DFS3499I ACTIVE DDNAMES: MODBLKSA IMSACBA  FORMATA  MODSTAT ID:   11
DFS3804I LAST CHKPT ID VALID FOR RESTART: 85135/161847-BUILDQ: 85135/161213


DFS994I TAKEOVER COMPLETED.
DFS3859I 16:18:29 PRIORITY 4 SESSIONS SWITCHED.
DFS3860I 16:18:29 ALL TERMINAL SESSIONS SWITCHED.
--------------------------------------------------------------------------------
     DATABASE
    DD41M803
        ERROR DD  TYPE    BLOCK
        DD41M803  IOT     0000003F
    *85135/161902*




                                                      PASSWORD:
/dis db dd41m803 bkerr
```

Figure 8. I/O Toleration Phase of Processing

Figure 9 on page 703 shows a screen of the use of the /UNLOCK SYSTEM command to notify the newly created active system that I/O prevention is complete (the XRF system status line now indicates that processing is no longer degraded by I/O toleration).

```
02/05/15 16:34:14  RSENAME: DFSRSENM  ACTIVE                          IMSB
DFS2716I NO MSDBS FOUND - NO MSDB CHECKPOINT TAKEN
 DFS994I *CHKPT 85135/161847**SIMPLE**
DFS3499I ACTIVE DDNAMES: MODBLKSA IMSACBA  FORMATA  MODSTAT ID:   11
DFS3804I LAST CHKPT ID VALID FOR RESTART: 85135/161847-BUILDQ: 85135/161213


 DFS058 16:34:14 UNLOCK COMMAND IN PROGRESS
DFS0488I - UNLOCK COMMAND COMPLETED.  RC = 00
DFS3860I 15:29:19 ALL TERMINAL SESSIONS SWITCHED.
--------------------------------------------------------------------------------









                                                      PASSWORD:
/unlock system
```

Figure 9. /UNLOCK SYSTEM Command

Figure 10 is a screen that shows that the I/O toleration EEQE for database DD41M803 has been deleted as part of `/UNLOCK SYSTEM` processing.

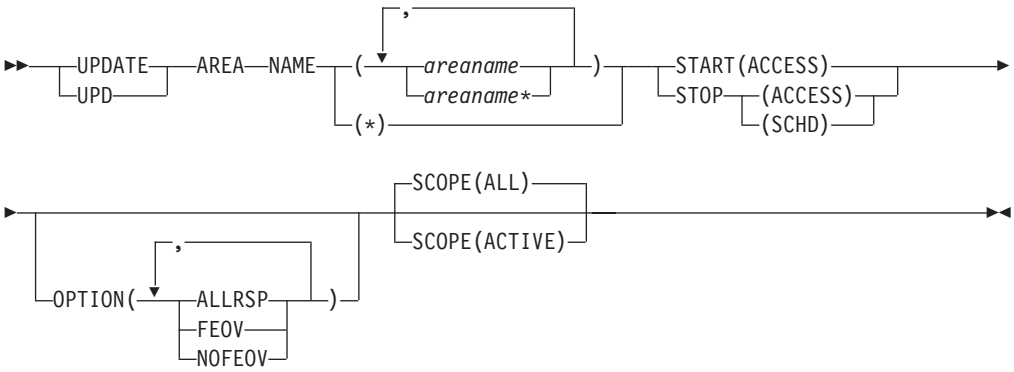
```
02/05/15 16:35:00  RSENAME: DFSRSENM  ACTIVE                            IMSB
DFS2716I NO MSDBS FOUND - NO MSDB CHECKPOINT TAKEN
 DFS994I *CHKPT 85135/161847**SIMPLE**
DFS3499I ACTIVE DDNAMES: MODBLKSA IMSACBA  FORMATA  MODSTAT ID:   11
DFS3804I LAST CHKPT ID VALID FOR RESTART: 85135/161847  -  BUILDQ: 85135/161213


DFS058 16:34:14 UNLOCK COMMAND IN PROGRESS
DFS0488I - UNLOCK COMMAND COMPLETED.  RC = 00
DFS3860I 16:18:29 ALL TERMINAL SESSIONS SWITCHED.
--------------------------------------------------------------------------------
     DATABASE
    DD41M803
        NO EEQE OR INCOMPLETE BACKOUT INFORMATION AVAILABLE
    *85135/163500*









                                                      PASSWORD:
/dis db dd41m803 bkerr
```

Figure 10. EEQE Deleted as Part of /UNLOCK SYSTEM Processing

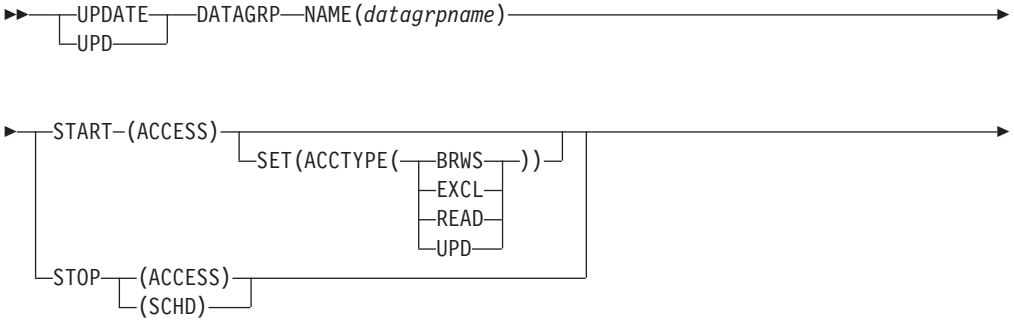
Chapter 65. UPDATE

Format

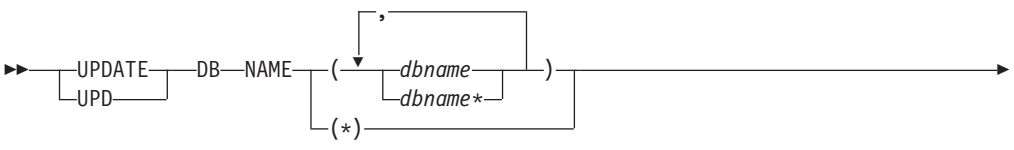
UPDATE AREA :



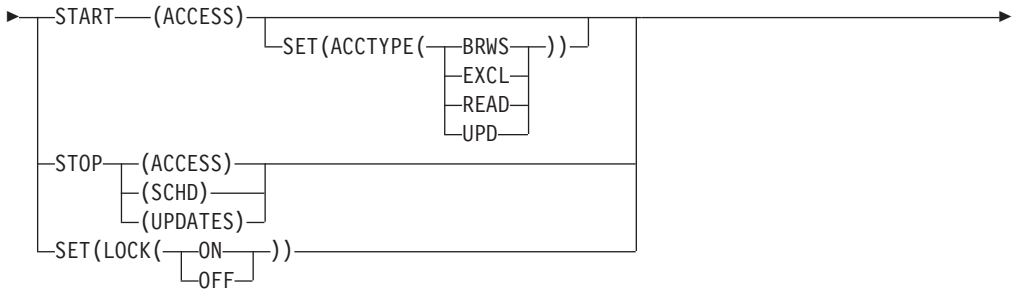
UPDATE DATAGRP:

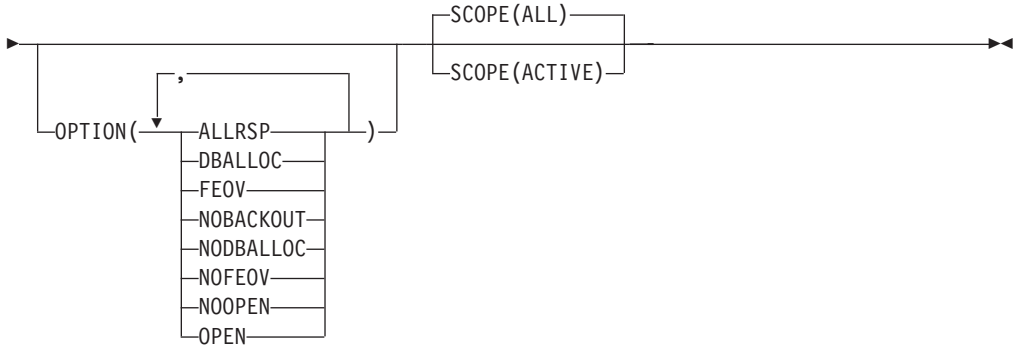


UPDATE DB:

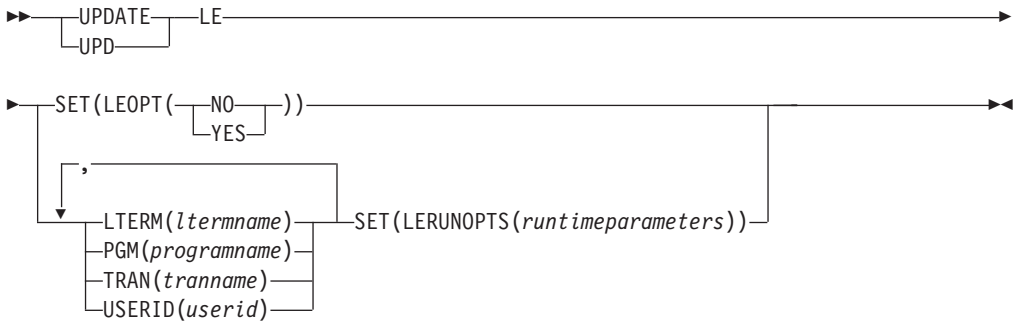


© Copyright IBM Corp. 1974, 2003

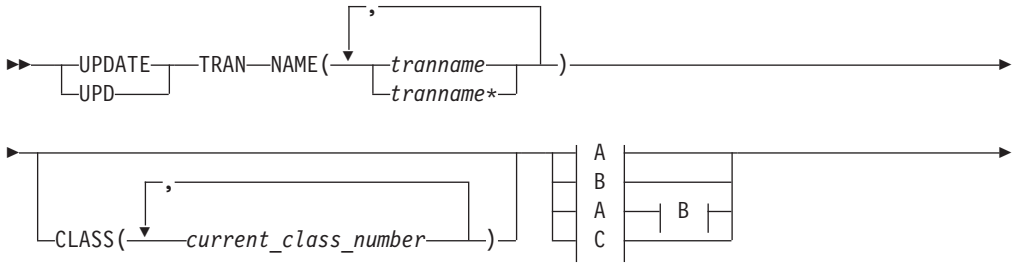




UPDATE LE:



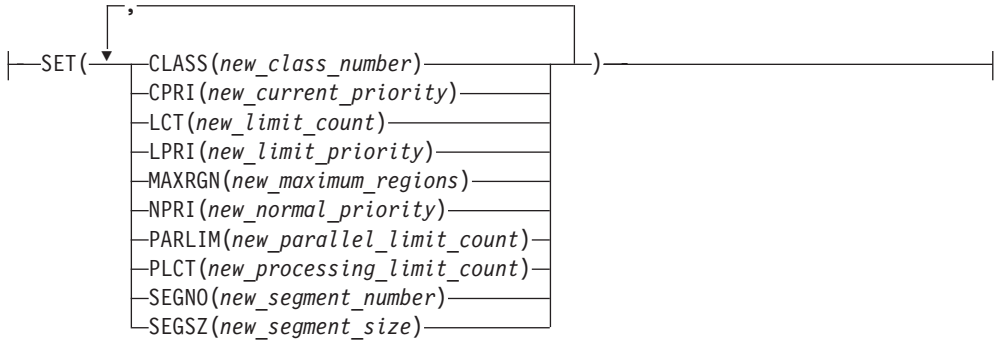
UPDATE TRAN:



A:

B:

C:



Environments and Keywords

Table 164, Table 165 on page 708, Table 166 on page 708, Table 167 on page 708, and Table 168 on page 708 list the environments (DB/DC, DBCTL, and DCCTL) from which the UPDATE command and keywords can be issued.

Table 164. Valid Environments for the UPDATE AREA Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
UPDATE AREA	X	X	
NAME	X	X	
OPTION	X	X	
SCOPE	X	X	
START	X	X	
STOP	X	X	

Table 165. Valid Environments for the UPDATE DATAGRP Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
UPDATE DATAGRP	X	X	
NAME	X	X	
OPTION	X	X	
SCOPE	X	X	
START	X	X	
STOP	X	X	

Table 166. Valid Environments for the UPDATE DB Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
UPDATE DB	X	X	
NAME	X	X	
OPTION	X	X	
SCOPE	X	X	
SET	X	X	
START	X	X	
STOP	X	X	

Table 167. Valid Environments for the UPDATE LE Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
UPDATE LE	X	X	X
LTERM	X	X	X
PGM	X	X	X
SET	X	X	X
TRAN	X	X	X
USERID	X	X	X

Table 168. Valid Environments for the UPDATE TRAN Command and Keywords

Command / Keywords	DB/DC	DBCTL	DCCTL
UPDATE TRAN	X		X
CLASS	X		X
CPRI	X		X
LCT	X		X
LPRI	X		X
MAXRGN	X		X
NAME	X		X
NPRI	X		X
PARLIM	X		X
PLCT	X		X
SCOPE	X		X
SEGNO	X		X

Table 168. Valid Environments for the UPDATE TRAN Command and Keywords (continued)

Command / Keywords	DB/DC	DBCTL	DCCTL
SEGSZ	X		X
SET	X		X
START	X		X
STOP	X		X

UPDATE AREA

The UPDATE AREA command allows changes to the status of area resources. The UPDATE AREA command can only be specified using the OM API and is processed only by the DB/DC and DBCTL environments. The command is also not allowed on the XRF alternate.

When the UPDATE AREA command is issued, the command only applies to the IMS to which it is routed. The command does not preload or preopen areas on other IMSs in the IMSplex that share the area. UPDATE AREA is routed by OM. OM routes the command to all active DB/DC or DBCTL IMS systems, unless specific routing is specified. OM selects one IMS as the command master.

The UPDATE AREA STOP(SCHD) command is not allowed on any RSR tracker. However, the commands, UPDATE AREA START(ACCESS) or UPDATE AREA STOP(ACCESS), are allowed only on a database level RSR tracker. UPDATE AREA START(ACCESS) and UPDATE AREA STOP(SCHD) are recoverable and a X'22' log record is written but UPDATE AREA STOP(ACCESS) is not recoverable.

NAME()

Specifies the names of the specific areas that are to be processed or a group of areas to be processed whose names match the generic or wild card parameter. NAME(*) indicates that the command is to be applied to all the areas in the system.

Wildcard parameters can be specified. The area names that match the generic or wildcard parameter are processed. For specific or wildcard names, response lines are returned for all the area names that are processed.

When the NAME(*) is specified, the response lines are returned for only the area names that resulted in an error. If OPTION(ALLRSP) is specified with NAME(*), response lines are returned for all the area names that are processed.

OPTION()

Specifies the additional functions to be performed. Following is a list of additional functions:

ALLRSP

Indicates that the response lines are to be returned for all areas that are processed on the command. ALLRSP is only valid with NAME(*). The default action is to return response lines only for the areas that resulted in an error.

FEOV Indicates to force end of volume after the command processing is complete. The IMS log switches to the next OLDS and a simple checkpoint is taken. This switch is marked as a recovery point for log archiving purposes.

FEOV is only valid with STOP(ACCESS) and is not valid on an RSR tracker.

NOFEOV
Indicates to not force end of volume after the command processing is complete. The IMS log does not switch to the next OLDS and a simple checkpoint is not taken.

OPTION(NOFEOV) is the default. NOFEOV is only valid with STOP(ACCESS).

SCOPE()
Specifies where IMS should apply changes. The default is ALL.

ACTIVE
SCOPE(ACTIVE) specifies that the changes are to be applied to the IMS systems that are currently active and to which the command is routed.

ALL SCOPE(ALL) specifies that the changes are to be applied to the active IMS systems.

ACTIVE and ALL produce the same results.

START()
Specifies the attributes that are to be started.

ACCESS
Specifies the specific areas of a DEDB to be allocated or reallocated.

An UPDATE AREA START(ACCESS) can be issued on an RSR tracker to resume tracking for those areas that were stopped or had tracking errors. In addition, Online Forward Recovery (OFR) is started for areas that are not current with mainline tracking.

For virtual storage option (VSO) areas, the UPDATE AREA START(ACCESS) can be used to preopen areas that are defined with the PREOPEN option. For VSO areas defined with the PRELOAD option, the command causes the areas to be loaded into the z/OS data space or an XES structure depending on the sharelevel of the area. The command has no effect on VSO areas that are in virtual storage.

STOP()
Specifies the attributes that are to be stopped.

ACCESS
Stops the access and updating of the specified DEDB areas and closes them.

The UPDATE AREA STOP(ACCESS) command for VSO areas removes the areas from the data space or XES structure and forces updates to be written back to DASD.

An UPDATE AREA START(ACCESS) command is required to open and reallocate the areas closed by the UPDATE AREA STOP(ACCESS) command.

SCHD Specifies that the data sets associated to the areas are to be closed and deallocated.

Note: An UPDATE AREA START(ACCESS) is required to reallocate the areas stopped by the UPDATE AREA STOP(SCHD) command.

If UPDATE AREA STOP(SCHD) is processed during HSSP processing, the area will be released after the current commit processing completes. Any image copy option in process when the command is issued can affect the continued system operation.

For VSO DEDB areas that are in a z/OS data space or XES structure, the UPDATE AREA STOP(SCHD) command action results in the removal of the VSO areas from the data space or XES structure and having the updates written to DASD.

UPDATE AREA Output Fields

Table 169 shows the UPDATE AREA output fields. The columns in the table are as follows:

Short Label
Contains the short label generated in the XML output.

Keyword
Identifies keyword on the command that caused the field to be generated. N/A appears for output fields that are always returned. ERR appears for output fields that are returned only in case of an error.

Meaning
Provides a brief description of the output field.

Table 169. Output Fields for the UPDATE AREA Command

Short Label	Keyword	Meaning
AREA	ERR	Area name. The area name is always returned.
CC	N/A	Completion code for the line of output. Completion code is always returned.
EERT	ERR	Error text returned to add more meaning to the completion code and may include a return code from a service. The error text may be returned for a non-zero completion code.
MBR	N/A	The IMSplex member that built the output line. The IMS identifier of the IMS for which the area information is displayed. The IMS identifier is always returned.

Return, Reason, and Completion Codes for UPDATE AREA

An IMS return and reason code is returned to OM by the UPDATE AREA command. The OM return and reason codes that may be returned as a result of the UPDATE AREA command are standard for all commands entered through the OM API. See *IMS Version 9: Common Service Layer Guide and Reference* for a list of the OM codes and the code meanings.

Table 170 on page 712 includes the return and reason codes and a brief explanation of the UPDATE AREA command. Table 171 on page 713 includes an explanation of the completion codes. Errors unique to the processing of UPDATE AREA command are returned as completion codes. A completion code is returned for each action against an individual resource.

Table 170. Return and Reason Codes for the UPDATE AREA Command

Return Code	Reason Code	Meaning
X'00000000'	X'000000000'	The UPDATE AREA command completed successfully.
X'00000008'	X'00002014'	The UPDATE AREA command is not processed because an invalid character is found in the area name parameter.
X'00000008'	X'00002040'	More than one filter or keyword value is specified on the UPDATE AREA command. Either more than one keyword or an invalid combination of filters was specified. For example, START(ACCESS) and STOP(ACCESS) was specified, or OPTION(NOFEOV,FEOV) was specified. Check the input command and reenter the correct combinations.
X'0000000C'	X'00003000'	The UPDATE AREA command is successful for at least one resource name. The UPDATE AREA command was not successful for one or more resource names. The completion code indicates the reason for the error with the resource name. The completion codes that can be returned by the UPDATE AREA command are listed in Table 171 on page 713.
X'0000000C'	X'00003004'	The UPDATE AREA command was not successful for all the resource name(s) specified. The completion code indicates the reason for the error with the resource name. The completion codes that can be returned by the UPDATE AREA command are listed in Table 171 on page 713.
X'00000010'	X'0000400C'	The UPDATE AREA command is not processed because the flavor entered is not valid on the XRF alternate.
X'00000010'	X'00004014'	The UPDATE AREA command is not processed because the function is not valid on the RSR tracker.
X'00000010'	X'00004024'	The UPDATE AREA command is not processed because Fast Path is not installed.
X'00000014'	X'00005004'	The UPDATE AREA command processing terminated because a DFSCOMD response buffer could not be obtained.
X'00000014'	X'00005008'	The UPDATE AREA command processing terminated because a DFSPOOL storage could not be obtained.
X'00000014'	X'0000500C'	The UPDATE AREA command processing terminated because AWE could not be obtained.
X'00000014'	X'00005014'	The UPDATE AREA command processing terminated because a WKAP storage could not be obtained.
X'00000014'	X'00005FFF'	The UPDATE AREA command processing terminated due to an internal error.

Table 171. Completion Codes for the UPDATE AREA Command

Completion Code	Meaning
0	The command completed successfully for the resource.
10	No resource found.
11	Duplicate resource name. The resource name is specified multiple times on the command and is ignored.
55	No Fast Path installed. The command failed because Fast Path is not installed.
90	Internal error. The command entered is not processed due to an internal error.
92	Command processing error. The command entered is not processed due to a command error. A unique completion code could not be generated to explain the error. The message number and the return code that could not be converted to a completion code are listed in the error text.
A5	Prevent Further Auth ON. The command entered is not processed because the database or area is defined to DBRC as prevent further Auth.
E0	Database or area in recovery. The command is not processed because the database or area is in recovery.
F0	No AREA lock. The command processing failed because the area lock could not be obtained.
F1	Area not stopped. The command entered is not processed because the AREA is not stopped.
F2	Preload is active for AREA. The command is not processed because preload is active for the AREA.
F3	Unresolved indoubts for AREA. The command entered is not processed because unresolved indoubts exist for the AREA.
F4	Allocation failed. Allocation failed for the AREA name.
F5	AREA needs recovery. The command processing failed because the area needs recovery.

Table 171. Completion Codes for the UPDATE AREA Command (continued)

Completion Code	Meaning
F6	ADS number discrepancy. The command failed for the AREA because there is a discrepancy between the number of ADS allocated by IMS and the number of ADS known to DBRC. Correct the discrepancy and reissue the command to deallocate the data sets.
F7	Area is not loaded into CF. The AREA OPEN failed and is not loaded into the Coupling Facility.
F8	Area has I/O tolerated CI. The command is not processed because the AREA has a I/O tolerated CI.
F9	Area has second CI EEQE. The command is not processed because the AREA has a second CI EEQE.

The UPDATE AREA Command Compared to Other Commands

Table 172 shows different instances of the `UPDATE AREA` command and other IMS commands that perform similar functions.

Table 172. UPDATE AREA Command Compared to Other Similar IMS Commands

UPDATE AREA Command	Similar IMS Commands
`UPDATE AREA NAME(name) STOP(ACCESS)`	`/DBR AREA areaname`
`UPDATE AREA NAME(name) START(ACCESS)`	`/START AREA areaname`
`UPDATE AREA NAME(name) STOP(SCHD)`	`/STOP AREA areaname`

Examples for UPDATE AREA Command

The following are examples of the `UPDATE AREA` command:

Example 1 for UPDATE AREA Command

TSO SPOC input:

```
UPD AREA NAME(DB21AR1*) STOP(ACCESS)
```

TSO SPOC output:

```
AreaName MbrName    CC
DB21AR1  IMS2        0
DB21AR1  SYS3        0
DB21AR10 IMS2        0
DB21AR10 SYS3        0
DB21AR11 IMS2        0
DB21AR11 SYS3        0
```

OM API input:

```
CMD(UPD AREA NAME(DB21AR1*) STOP(ACCESS))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.2.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2003.132 16:17:33.260435</statime>
<stotime>2003.132 16:17:33.281165</stotime>
<staseq>B968A333F5A93283</staseq>
<stoseq>B968A333FAB8DC83</stoseq>
<rqsttkn1>USRT005 10091733</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<master>IMS2    </master>
<userid>USRT005 </userid>
<verb>UPD </verb>
<kwd>AREA            </kwd>
<input>UPD AREA NAME(DB21AR1*) STOP(ACCESS) </input>
</cmd>
<cmdrsphdr>
<hdr slbl="AREA" llbl="AreaName" scope="LCL" sort="a" key="1"
 scroll="no" len="8" dtype="CHAR" align="left" skipb="no"  />
<hdr slbl="MBR" llbl="MbrName" scope="LCL" sort="a" key="2" scroll="no"
 len="8" dtype="CHAR" align="left" skipb="no" />
<hdr slbl="CC" llbl="CC" scope="LCL" sort="n" key="0" scroll="yes"
 len="4" dtype="INT" align="right" skipb="no" />
<hdr slbl="ERRT" llbl="ErrorText" scope="LCL" sort="n" key="0"
 scroll="yes" len="16" dtype="CHAR" align="left" skipb="yes" />
</cmdrsphdr>
<cmdrspdata>
<rsp>AREA(DB21AR1 ) MBR(IMS2    ) CC(   0) </rsp>
<rsp>AREA(DB21AR10) MBR(IMS2    ) CC(   0) </rsp>
<rsp>AREA(DB21AR11) MBR(IMS2    ) CC(   0) </rsp>
<rsp>AREA(DB21AR1 ) MBR(SYS3    ) CC(   0) </rsp>
<rsp>AREA(DB21AR10) MBR(SYS3    ) CC(   0) </rsp>
<rsp>AREA(DB21AR11) MBR(SYS3    ) CC(   0) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: The command stops access to all the areas that match the area name specified and makes them unavailable.The areas data sets are closed and deallocated.

Example 2 of UPDATE AREA Command

TSO SPOC input:

```
UPD AREA NAME(*) STOP(SCHD)
```

TSO SPOC output:

```
Log for . . : UPD AREA NAME(*) STOP(SCHD)
 IMSplex . . . . . : PLEX1
 Routing . . . . . :
 Start time. . . . : 2003.132 09:22:15.79
 Stop time . . . . : 2003.132 09:22:17.53
 Return code . . . : 00000000
 Reason code . . . : 00000000
 Command master. .  IMS2
```

OM API input:

```
CMD(UPD AREA NAME(*) STOP(SCHD))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.2.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2003.132 16:22:15.799896</statime>
<stotime>2003.132 16:22:17.536107</stotime>
<staseq>B968A44169058C8E</staseq>
<stoseq>B968A44310E6BE2D</stoseq>
<rqsttkn1>USRT005 10092215</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<master>IMS2    </master>
<userid>USRT005 </userid>
<verb>UPD </verb>
<kwd>AREA            </kwd>
<input>UPD AREA NAME(*) STOP(SCHD) </input>
</cmd>
</imsout>
```

Explanation: The command is routed to IMS2 and SYS3 and the command is successful at both IMSs. No response lines are returned because the default action for NAME(*) is to return response lines only for the area names that resulted in an error. OPTION(ALLRSP) can be specified to obtain all the area names processed on the command.

UPDATE DATAGRP

The UPDATE DATAGRP command allows changes to the status of the members of a data group. The command can only be specified through the OM API and can be processed only by the DB/DC and DBCTL environments. In addition, the UPDATE DATAGRP command is not allowed on the XRF alternate.

When the UPDATE DATAGRP command is issued, it only applies to the IMS system to which it is routed. It does not apply to the other IMS systems in the IMSplex that share the database or areas in the data group. The routing of the UPDATE DATAGRP command is done by OM. OM routes the command to all active DB/DC or DBCTL IMS systems, unless specific routing is specified. OM selects one IMS as the command master.

A response line is returned for the data group name from each IMS. Response lines are also returned for each data group member that resulted in an error. No response lines are returned for the data group members that are processed successfully unless OPTION(ALLRSP) is specified.

NAME()
Specifies the name of the data group to be processed.

OPTION()
Specifies the additional functions to be performed. Following is a list of additional functions:

ALLRSP
Indicates that the response lines are to be returned for all the members of the datagroup (all databases, areas, or both) that are processed on the command.

The default action is to return response lines only for the databases and areas or both that resulted in an error. A response line with the datagroup name is always returned.

DBALLOC
Indicates that the databases in the data group are to be allocated when they are started.

DBALLOC can only be specified with `START(ACCESS)`.

Only one of DBALLOC or NODBALLOC can be specified.

FEOV Indicates to force end of volume after the command processing is complete. The IMS log switches to the next OLDS and a simple checkpoint is taken. This switch is marked as a recovery point for log archiving purposes.

FEOV is only valid with `STOP(ACCESS)`. FEOV is not valid on an RSR tracker.

NODBALLOC
Indicates that the databases in the data group are not to be allocated when they are started. The databases will be allocated when they are scheduled.

NODBALLOC can only be specified with `START(ACCESS)`. NODBALLOC is the default action for `UPDATE DATAGRP` command if `OPTION(DBALLOC)` is not specified.

Only one of DBALLOC or NODBALLOC can be specified.

NOFEOV
Indicates to not force end of volume after the command processing is complete. The IMS log does not switch to the next OLDS and a simple checkpoint is not taken.

`OPTION(NOFEOV)` is the default action after the `UPDATE DATAGRP STOP(ACCESS)` command is processed.

NOFEOV is only valid with `STOP(ACCESS)`.

SCOPE()
Specifies where IMS should apply the change.

ACTIVE
`SCOPE(ACTIVE)` specifies that the changes are to be applied to the IMS systems that are currently active and to which the command is routed.

ALL `SCOPE(ALL)` specifies that the changes are to be applied to the active IMS systems.

Currently, ACTIVE and ALL produce the same results.

SET() Specifies the attribute values to be changed or sets the database state.

ACCTYP
Specifies the access intent for the named database. This keyword can be specified only if `START(ACCESS)` is specified. This keyword overrides the database access intent specified at system definition.

BRWS
Specifies that the named database is available for read-only processing on this IMS subsystem. The only programs that

can use the database on this subsystem are those that have a PCB processing option of GO (PROCOPT=GO). Programs that access the data using the GO processing option might see uncommitted data because a sharing IMS subsystem may be updating the database. The database is opened for read-only.

EXCL Specifies that the named database is to be used exclusively by this IMS subsystem. This exclusive access is guaranteed only when the database is registered to DBRC.

READ Specifies that the named database is available for read-only processing in this IMS subsystem. Programs with update intent can be scheduled, but cannot update the database. ACCTYP of READ differs from ACCTYP of BRWS in that the data is read with integrity(locking is performed) and all programs can access the data, not just those with a processing option of GO. The database is opened for read-only.

UPD Specifies that the named database is available for update as well as read processing in this IMS subsystem.

OFFLINE
Specifies that offline processing is to be performed for the members of the specified data group.

START()
Specifies the attributes that are to be started.

ACCESS
Specifies that the members of the specified data group name are to be started. See `UPDATE AREA START(ACCESS)` and `UPDATE DB START(ACCESS)` for a description of the actions performed on the databases and areas in the data group.

Additional functions to be peformed along with `START(ACCESS)` can be specified using the OPTION keyword.

An `UPDATE DATAGRP START(ACCESS)` command with `SET(ACCTYP)` or `OPTION(DBALLOC|NODBALLOC|NOBACKOUT|OPEN|NOOPEN)` is invalid on the RSR tracker.

STOP()
Specifies the attributes to be stopped.

ACCESS
Specifies that offline processing is to be done for the members of the specified datagroup. See `UPDATE AREA STOP(ACCESS)`and `UPDATE DB STOP(ACCESS)` for a description of the actions performed on the databases and areas in the data group.

SCHD Specifies that the members of the specified data group name are to be stopped. See `UPDATE AREA STOP(ACCESS)` and `UPDATE DB STOP(ACCESS)` for a description of the actions performed on the databases and areas in the data group.

UPDATE DATAGRP Output Fields

Table 173 on page 719 shows the output fields for the `UPDATE DATAGRP`. The columns in the table are as follows:

Short Label
Contains the short label generated in the XML output.

Keyword
Identifies the keyword on the command that caused the field to be generated. N/A appears for output fields that are always returned. ERR appears for output fields that are returned only in case of an error.

Meaning
Provides a brief description of the output field.

Table 173. Output Fields for UPDATE DATAGRP Command

Short Label	Keyword	Meaning
AREA	N/A	Area name. The Area name is returned if there are one or more areas in the data group.
CC	N/A	Completion code for the line of output. Completion code is always returned.
DB	N/A	Database name. The database name is returned if there are one or more databases in the data group.
DG	N/A	Data group name. The data group name is always returned.
ERRT	ERR	Error text returned to add more meaning to the completion code and may include a return code from a service. The error text is only returned if the completion code is non-zero.
MBR	N/A	The IMSplex member that built output line. The IMS identifier of the IMS for which the database information is displayed. The IMS identifier is always returned.

Return, Reason, and Completion Codes for the UPDATE DATAGRP Command

An IMS return and reason code is returned to OM by the UPDATE DATAGRP command. The OM return and reason codes that may be returned as a result of the UPDATE DATAGRP command are standard for all commands entered through the OM API. See *IMS Version 9: Common Service Layer Guide and Reference* for a list of the OM codes and the code meanings.

Table 174 includes the return and reason codes and a brief explanation of the UPDATE DATAGRP command. Table 175 on page 720 includes an explanation of the completion codes. Errors unique to the processing of UPDATE DATAGRP command are returned as completion codes. A completion code is returned for each action against an individual resource.

Table 174. Return and Reason Codes for the UPDATE DATAGRP Command

Return Code	Reason Code	Meaning
X'00000000'	X'00000000'	The UPDATE DATAGRP command completed successfully.
X'00000008'	X'00002004'	An invalid keyword or more than one keyword is specified on the UPDATE DATAGRP command.
X'00000008'	X'00002014'	The UPDATE DATAGRP command is not processed because an invalid character is found in the data group name.

Table 174. Return and Reason Codes for the UPDATE DATAGRP Command (continued)

Return Code	Reason Code	Meaning
X'00000008'	X'00002040'	More than one filter or keyword value is specified on the UPDATE DATAGRP command. Either more than one keyword or an invalid combination of filters was specified. For example, START(ACCESS) and STOP(ACCESS) was specified, or OPTION(DBALLOC,NODBALLOC) was specified. Check the input command and reenter the correct combinations.
X'0000000C'	X'00003000'	THe UPDATE DATAGRP command is successful for at least one member in the data group. The UPDATE DATAGRP command is not successful for one or more members in the data group. The completion code indicates the reason for the error with the data group member name. The completion codes that can be returned by the UPDATE DATAGRP command are listed in the UPDATE DATAGRP completion code table.
X'0000000C'	X'00003004'	The UPDATE DATAGRP command is not successful for all the members in the data group. The completion code indicates the reason for the error with the data group member name. The completion codes that can be returned by the UPDATE DATAGRP command are listed in the UPDATE DATAGRP completion code table.
X'00000010'	X'0000400C'	The UPDATE DATAGRP command is not processed because the flavor entered is not valid on the XRF alternate.
X'00000010'	X'00004014'	The UPDATE DATAGRP command is not processed because the function is not valid on the RSR tracker.
X'00000014'	X'00005004'	The UPDATE DATAGRP command processing terminated because a DFSOCMD response buffer could not be obtained.
X'00000014'	X'00005008'	The UPDATE DATAGRP command processing terminated because DFSPOOL storage could not be obtained.
X'00000014'	X'0000500C'	The UPDATE DATAGRP command processing terminated because AWE could not be obtained.
X'00000014'	X'00005014'	The UPDATE DATAGRP command processing terminated because a WKAP storage could not be obtained.
X'00000014'	X'00005FFF'	The UPDATE DATAGRP command processing terminated due to an internal error.

Table 175. Completion Codes for the UPDATE DATAGRP Command

Completion Code	Meaning
	See UPDATE AREA completion code table (Table 171 on page 713) for codes that can be returned for the areas in a data group.
	See UPDATE DB completion code table (Table 179 on page 731) for codes that can be returned for the databases in the data group.

UPDATE DATAGRP Compared to Other Commands

Table 176 on page 721 shows different instances of the UPDATE DATAGRP command and other IMS commands that perform similar functions.

Table 176. UPDATE DATAGRP Command Compared to Other IMS Commands

UPDATE DATAGRP Command	Similar IMS Commands
UPDATE DATAGRP NAME(name) STOP(ACCESS)	/DBR DATAGRP datagrpname
UPDATE DATAGRP NAME(name) START(ACCESS)	/START DATAGRP datagrpname
UPDATE DATAGRP NAME(name) STOP(SCHD)	/STOP DATAGRP datagrpname

Examples for UPDATE DATAGRP Command

The following are examples of the UPDATE DATAGRP command:

Example 1 for UPDATE DATAGRP Command

TSO SPOC input:

```
UPD DATAGRP NAME(GROUP1) STOP(ACCESS)
```

TSO SPOC output:

```
DataGroup DBName   AreaName MbrName    CC
GROUP1                      IMSA        0
GROUP1                      IMS1        8
GROUP1    DEDBJN03          IMS1        C
GROUP1    DEDBJN03 DB3AREA0 IMS1       F0
```

OM API input:

```
CMD(UPD DATAGRP NAME(GROUP1) STOP(ACCESS))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.2.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2003.129 22:45:55.898843</statime>
<stotime>2003.129 22:45:55.990001</stotime>
<staseq>B965346AB45DB229</staseq>
<stoseq>B965346ACA9F1263</stoseq>
<rqsttkn1>USRT005 10154555</rqsttkn1>
<rc>0200000C</rc>
<rsn>00003000</rsn>
</ctl>
<cmderr>
<mbr name="IMS1    ">
<typ>IMS     </typ>
<styp>DBDC    </styp>
<rc>0000000C</rc>
<rsn>00003000</rsn>
</mbr>
</cmderr>
<cmd>
<master>IMSA    </master>
<userid>USRT005 </userid>
<verb>UPD </verb>
<kwd>DATAGRP         </kwd>
<input>UPD DATAGRP NAME(GROUP1) STOP(ACCESS) </input>
</cmd>
<cmdrsphdr>
<hdr slbl="DG" llbl="DataGroup" scope="LCL" sort="a" key="1"
 scroll="no" len="8" dtype="CHAR" align="left" skipb="no" />
<hdr slbl="DB" llbl="DBName" scope="LCL" sort="a" key="3" scroll="no"
 len="8" dtype="CHAR" align="left" skipb="yes" />
<hdr slbl="AREA" llbl="AreaName" scope="LCL" sort="a" key="4"
 scroll="no" len="8" dtype="CHAR" align="left" skipb="yes" />
```

```
<hdr slbl="MBR" llbl="MbrName" scope="LCL" sort="a" key="2" scroll="no"
 len="8" dtype="CHAR" align="left" skipb="no" />
<hdr slbl="CC" llbl="CC" scope="LCL" sort="n" key="0" scroll="yes"
 len="4" dtype="INT" align="right" skipb="no" />
<hdr slbl="ERRT" llbl="ErrorText" scope="LCL" sort="n" key="0"
 scroll="yes" len="16" dtype="CHAR" align="left" skipb="yes" />
</cmdrsphdr>
<cmdrspdata>
<rsp>DG(GROUP1  ) MBR(IMSA    ) CC(   0) </rsp>
<rsp>DB(DEDBJN03) AREA(DB3AREA0) DG(GROUP1  ) MBR(IMS1    ) CC(  F0)
</rsp>
<rsp>DB(DEDBJN03) DG(GROUP1  ) MBR(IMS1    ) CC(   C) </rsp>
<rsp>DG(GROUP1  ) MBR(IMS1    ) CC(   8) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: The command stops access to all the members of datagroup GROUP1 and takes them offline. The command response lines are returned for all the databases or areas that resulted in an error. No response lines are returned for members for which the command was successful.

Example 2 for UPDATE DATAGRP Command

TSO SPOC input:

```
UPD DATAGRP NAME(GROUP1) START(ACCESS) OPTION(ALLRSP)
```

TSO SPOC output:

```
DataGroup DBName   AreaName MbrName    CC
GROUP1                      IMSA        0
GROUP1            DB21AR1   IMSA        0
GROUP1    BE3PARTS          IMSA        0
GROUP1    DEDBJN03          IMSA        0
GROUP1                      IMS1        0
GROUP1            DB21AR1   IMS1        0
GROUP1    BE3PARTS          IMS1        0
GROUP1    DEDBJN03          IMS1        0
```

OM API input:

```
CMD(UPD DATAGRP NAME(GROUP1) START(ACCESS) OPTION(ALLRSP))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.2.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2003.129 22:46:51.490484</statime>
<stotime>2003.129 22:46:51.672332</stotime>
<staseq>B965349FB88B4445</staseq>
<stoseq>B965349FE4F0C36A</stoseq>
<rqsttkn1>USRT005 10154651</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<master>IMSA    </master>
<userid>USRT005 </userid>
<verb>UPD </verb>
<kwd>DATAGRP         </kwd>
<input>UPD DATAGRP NAME(GROUP1) START(ACCESS) OPTION(ALLRSP) </input>
</cmd>
<cmdrsphdr>
<hdr slbl="DG" llbl="DataGroup" scope="LCL" sort="a" key="1"
 scroll="no" len="8" dtype="CHAR" align="left" skipb="no" />
```

```
<hdr slbl="DB" llbl="DBName" scope="LCL" sort="a" key="3" scroll="no"
 len="8" dtype="CHAR" align="left" skipb="yes" />
<hdr slbl="AREA" llbl="AreaName" scope="LCL" sort="a" key="4"
 scroll="no" len="8" dtype="CHAR" align="left" skipb="yes" />
<hdr slbl="MBR" llbl="MbrName" scope="LCL" sort="a" key="2" scroll="no"
 len="8" dtype="CHAR" align="left" skipb="no" />
<hdr slbl="CC" llbl="CC" scope="LCL" sort="n" key="0" scroll="yes"
 len="4" dtype="INT" align="right" skipb="no" />
<hdr slbl="ERRT" llbl="ErrorText" scope="LCL" sort="n" key="0"
 scroll="yes" len="16" dtype="CHAR" align="left" skipb="yes" />
</cmdrsphdr>
<cmdrspdata>
<rsp>DB(DEDBJN03) DG(GROUP1  ) MBR(IMSA    ) CC(   0) </rsp>
<rsp>DB(BE3PARTS) DG(GROUP1  ) MBR(IMSA    ) CC(   0) </rsp>
<rsp>AREA(DB21AR1 ) DG(GROUP1  ) MBR(IMSA    ) CC(   0) </rsp>
<rsp>DG(GROUP1  ) MBR(IMSA    ) CC(   0) </rsp>
<rsp>DB(DEDBJN03) DG(GROUP1  ) MBR(IMS1    ) CC(   0) </rsp>
<rsp>DB(BE3PARTS) DG(GROUP1  ) MBR(IMS1    ) CC(   0) </rsp>
<rsp>AREA(DB21AR1 ) DG(GROUP1  ) MBR(IMS1    ) CC(   0) </rsp>
<rsp>DG(GROUP1  ) MBR(IMS1    ) CC(   0) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: The command starts access of all the members of datagroup, GROUP1, and makes them available. The command response lines are returned for all the databases or areas that are processed with OPTION(ALLRSP) specified.

UPDATE DB

The UPDATE DB command allows changes to the status of database resources. For example, make a database available, take the database offline, stop scheduling, stop updates, lock, and unlock a database. The UPDATE DB command may only be specified through the OM API and can only be processed by the DB/DC and DBCTL environments. Also, the command is not allowed on the XRF alternate system.

When the UPDATE DB command is issued, it only applies to the IMS system to which it is routed. The command is not processed by other IMS systems in the IMSplex that share the database but do not receive the command. The routing of the UPDATE DB command is done by OM. OM routes the command to all active DB/DC or DBCTL IMS systems, unless specific routing is specified. OM selects one IMS as the command master.

The UPDATE DB SET(LOCK(ON)), UPDATE DB SET(LOCK(OFF)), UPDATE DB STOP(SCHD), and UPDATE DB STOP(UPDATES) commands are not allowed on an RSR tracker. The UPDATE DB START(ACCESS) with SET(ACCTYP) or OPTION(DBALLOC|NODBALLOC|NOBACKOUT|OPEN|NOOPEN) are also invalid on the RSR tracker.

3 The UPDATE DB command is valid for HALDB databases. If the command is issued
3 for a HALDB master, it applies to all the HALDB partitions. Each partition inherits
3 the access limitations of the HALDB master. However, the status condition is only
3 displayed on the HALDB master and not on the partitions. If the command is issued
3 on one HALDB partition, it applies only to that HALDB partition and not the HALDB
3 master.

NAME()

Specifies the names of the specific databases that are to be processed or indicates that the command is to be applied to all the databases in the system if NAME(*) is specified.

NAME(*) is the architected method on IMSPlex commands to allow the user to specify that the command applies to all the database resources.

Wildcard parameters can be specified on the NAME keyword. The database names that match the generic or wildcard parameter are processed. For specific or wildcard names, response lines are returned for all the database names that are processed.

For NAME(*), response lines are returned for all database names that resulted in an error. If OPTION(ALLRSP) is specified with NAME(*), response lines are returned for all the database names that are processed.

When the database specified is a DEDB, a response line is returned for the DEDB name. A response line is also returned for all the areas of a DEDB that resulted in errors. A response line is not returned for the areas of a DEDB for which the command action is successful.

OPTION()
Specifies additional functions to be performed along with the UPDATE DB command.

ALLRSP
Indicates that the response lines are to be returned for all databases that are processed on the command. The default action is to return response lines only for the databases that resulted in an error. It is only valid with NAME(*).

DBALLOC
Indicates that the database is to be allocated when it is started.

DBALLOC can only be specified with START(ACCESS). DBALLOC is the default action when specific database names are specified on the command. Only one of DBALLOC or NODBALLOC can be specified.

OPTION(DBALLOC) is not valid for a HALDB master.

FEOV Indicates to force end of volume after the command processing is complete.

The IMS log switches to the next OLDS and a simple checkpoint is taken. This switch is marked as a recovery point for log archiving purposes. FEOV is only valid with STOP(UPDATES) and STOP(ACCESS).

FEOV does not apply to an RSR tracker.

NOBACKOUT
Indicates to suppress backout restart for a database not registered in DBRC.

NOBACKOUT can only be specified with START(ACCESS) and is not valid with NAME(*). OPTION(NOBACKOUT) is not valid for a HALDB master.

NODBALLOC
Indicates that the database is not to be allocated when it is started.

The database will be allocated when it is scheduled. NODBALLOC can only be specified with START(ACCESS). NODBALLOC is the default action when NAME(*) is specified. Only one of DBALLOC or NODBALLOC can be specified. OPTION(NODBALLOC) is ignored for HALDB masters.

NOFEOV
Indicates to not force end of volume after the command processing is complete. The IMS log does not switch to the next OLDS and a simple checkpoint is not taken. NOFEOV is the default.

OPTION(NOFEOV) can be specified with STOP(ACCESS) and STOP(UPDATES).

NOOPEN
Indicates that the database is not to be opened when it is started.

NOOPEN can only be specified with START(ACCESS). NOOPEN is the default action unless the database has EEQE's.

Only one of OPEN or NOOPEN can be specified. OPTION(NOOPEN) is not valid for a HALDB master.

OPEN Indicates that the database is to be opened when it is started. OPEN can only be specified with START(ACCESS).

OPEN cannot be specified with NAME(*) or if NODBALLOC is specified. Only one of OPEN or NOOPEN can be specified. OPTION(OPEN) is not valid for a HALDB master.

SCOPE()
Specifies where IMS should apply the change.

ACTIVE
Specifies that the changes are to be applied to the IMS systems that are currently active and to which the command is routed.

ALL Specifies that the changes are to be applied to the active IMS systems.

Currently, ACTIVE and ALL produce the same results.

SET() Specifies the attribute values to be changed or sets the database state.

ACCTYP
Specifies the access intent for the named database. This keyword can be specified only if START(ACCESS) is specified. This keyword overrides the database access intent specified at system definition.

BRWS
Specifies that the named database is available for read-only processing on this IMS subsystem. The only programs that can use the database on this subsystem are those that have a PCB processing option of GO (PROCOPT=GO). Programs that access the data using the GO processing option might see uncommitted data since a sharing IMS subsystem could be updating the database. The database is opened for read only.

EXCL Specifies that the named database is to be used exclusively by this IMS subsystem. This exclusive access is guaranteed only when the database is registered to DBRC.

READ Specifies that the named database is available for read-only processing in this IMS subsystem. Programs with update intent can be scheduled, but cannot update the database. ACCTYP of READ differs from ACCTYP of BRWS in that the data is read with integrity (locking is performed) and all

programs can access the data, not just those with a processing option of GO. The database is opened for read-only.

UPD Specifies that the named database is for update as well as read processing in the IMS subsystem.

LOCK Locks and unlocks the specified database. NAME(*) cannot be specified with SET(LOCK(ON|OFF)).

ON Locks and prevents subsequently scheduled programs from accessing the database, without affecting currently scheduled programs. The database is not closed.

For a shared secondary index database, an UPDATE DB SET(LOCK(ON)) on the first secondary index or subsequent secondary indexes affects only the named database.

OFF Unlocks the specified databases and resets the effect of an UPDATE DB SET(LOCK(ON)) command. An UPDATE DB SET(LOCK(OFF)) on the first secondary index or subsequent secondary indexes affects only the named database.

START()

Specifies the attributes that are to be started.

ACCESS

The UPDATE DB START(ACCESS) command starts the database and permits access from transactions or programs. The UPDATE DB START(ACCESS) command resets the actions done by a prior UPDATE DB STOP(SCHD), or UPDATE DB STOP(UPDATE). The access intent is set to the database access intent specified at system definition unless SET(ACCTYPE) is specified.

The UPDATE DB START(ACCESS) command can be used to allocate or reallocate all databases other than DEDBs. For a DEDB, an UPDATE AREA command can be used to allocate or reallocate the DEDB areas.

For a DEDB, the UPDATE DB START(ACCESS) command also causes any unloaded randomizer, that was specified in the DBD source, to be reloaded.

3 When the UPDATE DB START(ACCESS) command is specified for a
3 HALDB partition, the partition is not allocated unless it has EEQEs,
3 the OPEN keyword is specified, or it was previously authorized but
3 not allocated. The action taken to allocate the data sets is
3 dependant on the status of the master database and its availability.

The UPDATE DB START(ACCESS) command may reset the USTOPPED status for transactions that are suspended. If the transaction is suspended and its processing program has access to the started database, the UPDATE DB START(ACCESS) command will result in the USTOPPED attribute being reset. Any messages on the suspend queue for that transaction will be transferred to the normal processing queue.

If there was a prior dynamic backout or emergency restart backout failure, then the UPDATE DB START(ACCESS) command will attempt to perform the backout again.

If the database is registered to DBRC, then DBRC is informed when batch backout is successfully executed, and the failing backout will not be attempted again when a UPDATE DB START(ACCESS) command is issued.

The UPDATE DB START(ACCESS) command is not processed for the databases being accessed by batch programs.

For a HIDAM database, the UPDATE DB START(ACCESS) command must be issued for both the index and the data area DBD. If a backout failure occurs for this database, the command causes the backout to be attempted again.

On an RSR tracker, the UPDATE DB START(ACCESS) command can be used to resume tracking for those databases that were stopped by a tracking subsystem processing. The command can also be used to start online forward recovery (OFR) for those databases that are not current with mainline tracking.

For shared secondary index databases, the UPDATE DB START(ACCESS) can be issued on the first secondary index or subsequent secondary indexes to undo the actions of the prior UPDATE DB STOP(SCHD) and UPDATE DB STOP(UPDATES) commands.

Additional functions can be performed with START(ACCESS) by specifying the OPTION keyword. OPTION(DBALLOC | NODBALLOC), OPTION(NOBACKOUT), OPTION(NOOPEN | OPEN) can be specified along with START(ACCESS). The OPTION keyword is not valid for a HALDB master.

STOP()

Specifies the attributes that are to be stopped.

ACCESS

Starts offline processing of the database. This closes and deallocates the database and unauthorizes the database to DBRC. An UPDATE DB START(ACCESS) command is required to reset the effect of an UPDATE DB STOP(ACCESS).

OPTION(NOFEOV) is the default action for STOP(ACCESS). The IMS log is not switched to the next OLDS and a simple checkpoint is not taken. OPTION(FEOV) can be specified on the command to switch to the next OLDS and take an IMS simple checkpoint.

The UPDATE DB STOP(ACCESS) command can be used on a database readiness level (DLT) tracker to take shadow areas and databases offline for image copy and recovery. The command can also be used to stop online forward recovery (OFR) in progress for the specified database.

The UPDATE DB STOP(ACCESS) command will not deallocate a data set if a VSAM data set hardware error occurred. For shared secondary index databases, an UPDATE DB STOP(ACCESS) on the first secondary index affects all databases sharing the secondary index dataset. An UPDATE DB STOP(ACCESS) command on the subsequent secondary indexes affects only the named database.

SCHD Stops or prevents subsequently scheduled programs from accessing the database, without affecting currently scheduled programs. The database is not closed.

An UPDATE DB START(ACCESS) command can be used to reset the effect of an UPDATE DB STOP(SCHD) command.

If the command is issued for a DEDB or MSDB, programs using the database will not be scheduled. For other databases, the programs will still be scheduled. If the INIT call was issued, however, a call against the database will result in either a 3308 pseudo abend or a BA status code.

When the UPDATE DB STOP(SCHD) command is issued for a database that is in use by an MPP region, the command is processed after the region completes processing the current message. After the current message processing is complete, the application program receives a QC status indicating no more messages even if there are messages to be processed.

When the UPDATE DB STOP(SCHD) command is issued for a database that is in use by a BMP region, the command is rejected.

In a DBCTL system, CCTL can specify LONG or SHORT when it schedules a PSB. When the UPDATE DB STOP(SCHD) command is issued for a database that is in use by a LONG thread, the command is rejected. When the command is issued for a database that is in use by a SHORT thread, the thread completes before the command is processed.

For a shared secondary index database, an UPDATE DB STOP(SCHD) on the first secondary index or subsequent secondary indexes affects only the named databases.

UPDATES

Stops or prevents transactions or programs from updating the specified DL/I database.

STOP(UPDATES) is not valid for DEDBs or MSDBs.

An UPDATE DB START(ACCESS) command is required to reset the effect of an UPDATE DB STOP(UPDATES) command.

When the UPDATE DB STOP(UPDATES) command is processed, the message processing regions using the specified database are terminated at the conclusion of processing their transactions in preparation to close the database and allow the databases to be opened input only. As the message processing regions terminate programs, the data sets of the database are closed.

OPTION(FEOV) forces the IMS log to switch to the next OLDS. This switch to the next OLDS is marked as a recovery point for log archiving purposes. IMS also issues a simple checkpoint. OPTION(NOFEOV), which is the default, overrides this action.

After the command is processed, the scheduling of transactions is resumed. No transactions will be allowed to update the specified databases. Programs with update intent will be scheduled, but update calls to DL/I databases will result in a 3303 pseudoabend, a BA, or BB status code. The pseudoabend or status codes appear only if the application program informed IMS through the INIT STATUS GROUPA or GROUPB call that it is prepared to accept status codes regarding data unavailability.

An UPDATE DB STOP(UPDATES) on the first secondary index affects all databases sharing the secondary index data set. An UPDATE DB STOP(UPDATES) on subsequent secondary indexes affects only the named database.

UPDATE DB Output Fields

Table 177 shows the output fields for the UPDATE DB . The columns in the table are as follows:

Short Label
Contains the short label generated in the XML output.

Keyword
Identifies the keyword on the command that caused the field to be generated. N/A appears for output fields that are always returned. ERR appears for output fields that are returned only in case of an error.

Meaning
Provides a brief description of the output field.

Table 177. UPDATE DB Output Fields

Short Label	Keyword	Meaning
AREA	ERR	Area name of the DEDB that resulted in an error during the processing of the command.
CC	N/A	Completion code for the line of output. Completion code is always returned.
DB	N/A	Database name. The database name is always returned.
ERRT	ERR	Error text returned to add more meaning to the completion code and may include a return code from a service. The error text is only returned if the completion coe is non-zero.
MBR	N/A	IMSplex member that built output line. IMS identifier of the IMS for which the database information is displayed. IMS identifier is always returned.

Return, Reason, and Completion Codes for the UPDATE DB Command

An IMS return and reason code is returned to OM by the UPDATE DB command. The OM return and reason codes that may be returned as a result of the UPDATE DB command are standard for all commands entered through the OM API. See *IMS Version 9: Common Service Layer Guide and Reference* for a list of the OM codes and the code meanings.

Table 178 includes the return and reason codes and a brief explanation of the UPDATE DB command. Table 179 on page 731 includes an explanation of the completion codes. Errors unique to the processing of UPDATE DB command are returned as completion codes. A completion code is returned for each action against an individual resource.

Table 178. Return and Reason Codes for UPDATE DB

Return Code	Reason Code	Meaning
X'00000000'	X'00000000'	The UPDATE DB command completed successfully.
X'00000008'	X'00002004'	An invalid keyword or more than one keyword is specified on the UPDATE DB command.

Table 178. Return and Reason Codes for UPDATE DB (continued)

Return Code	Reason Code	Meaning
X'00000008'	X'00002011'	The UPDATE DB command processing terminated because NAME(*) is not supported with one or more keywords specified.
X'00000008'	X'00002014'	The UPDATE DB command is not processed because an invalid character is found in the database name parameter.
X'00000008'	X'00002040'	More than one filter or keyword value is specified on the UPDATE DB command. Either more than one keyword or an invalid combination of filters was specified. For example, START(ACCESS) and STOP(ACCESS) was specified, or OPTION(DBALLOC,NODBALLOC) was specified. Check the input command and reenter the correct combinations.
X'0000000C'	X'00003000'	The UPDATE DB command is successful for at least one resource name. The UPDATE DB command is not successful for one or more resource names. The completion code indicates the reason for the error with the resource name. The completion codes that can be returned by the UPDATE DB command are listed in the UPDATE DB completion code table.
X'0000000C'	X'00003004'	The UPDATE DB command is not successful for all the resource name(s) specified. The completion code indicates the reason for the error with the resource name. The completion codes that can be returned by the UPDATE DB command are listed in the UPDATE DB completion code table.
X'00000010'	X'00004000'	The UPDATE DB command is not processed as restart is in progress.
X'00000010'	X'0000400C'	The UPDATE DB command as entered is not processed because the function is not valid on the XRF alternate.
X'00000010'	X'00004014'	The UPDATE DB command is not processed because the function is not valid on the RSR tracker.
X'00000010'	X'00004024'	The UPDATE DB command is not processed because Fast Path is not installed.
X'00000010'	X'000041F0'	The UPDATE DB command is not processed because an IMS Checkpoint is in progress.
X'00000010'	X'000041F4'	The UPDATE DB command is not processed because an MSDB Checkpoint is in progress.
X'00000010'	X'000041F8'	The UPDATE DB command is not processed because a takeover is in progress.
X'00000014'	X'00005004'	The UPDATE DB command processing terminated because a DFSOCMD response buffer could not be obtained.
X'00000014'	X'00005008'	The UPDATE DB command processing terminated because a DFSPOOL stroage could not be obtained.
X'00000014'	X'0000500C'	The UPDATE DB command processing terminated because an AWE could not be obtained.
X'00000014'	X'00005014'	The UPDATE DB command processing terminated because a WKAP storage could not be obtained.

Table 178. Return and Reason Codes for UPDATE DB (continued)

Return Code	Reason Code	Meaning
X'00000014'	X'00005FFF'	The UPDATE DB command processing terminated due to an internal error.
X'00000015'	X'00005014'	The UPDATE DB command processing terminated because BCB storage could not be obtained.

Table 179. Completion Codes for UPDATE DB

Completion Code	Meaning
0	The command completed successfully for the resource.
8	Some The command completed with error for some of the AREAS of the DEDB. Response lines for the area names in error are returned.
C	None The command completed with error for all the AREAs of the DEDB. Response lines for the area names in error are returned.
10	No resource found
11	Duplicate resource name. The resource name is specified multiple times on the command and is ignored.
31	Database is a DEDB The command entered is not valid for the DEDB in the IMS environment.
32	Database is a MSDB The command entered is not valid for the MSDB in the IMS environment.
33	Command invalid HALDB master The command OPTION is invalid for the HALDB master.
55	No Fast Path installed The command failed because Fast Path is not installed.
56	Command invalid on the RSR tracker due to LSO=Y option.
65	DMB pool storage error The command failed due to DMB pool storage request failure.
66	DMB pool full The command failed due to the DMB pool being full.
81	DBRC Error
90	Internal error The command entered is not processed due to an internal error.
92	Command processing error The command entered is not processed due to an error. A unique completion code could not be generated to explain the error. The message number and the return code that could not be converted to a completion code are listed in the error text.
A0	Dynamic allocation failed The command entered is not processed because the dynamic allocation failed for the DB.
A1	Database is authorized by Batch The command entered is not processed because the database is authorized by batch.

Table 179. Completion Codes for UPDATE DB (continued)

Completion Code	Meaning
A2	Database is authorized by another IMS The command entered is not processed because the database is authorized by another active or abnormally terminated IMS and its authorization state is incompatible with the current authorization request.
A3	Authorization change failed The DBRC CHNGAUTH request resulted in an error.
A4	Database not registered to DBRC The command processing failed as the database is not registered to DBRC.
A5	Prevent further Auth ON The command entered is not processed because the database or area is defined to DBRC as prevent further Auth.
A6	Invalid database record in RECON The command entered is not processed because an invalid parameter was found during the evaluation process of the database usage compatibility. The database record might be invalid in the RECON dataset.
A7	DBRC unauth failed during change authorization The command is not processed due to an error during UNAUTH processing during change authorization request.
A8	Invalid database record in RECON An `UPDATE DB SET(ACCTYP)` command is entered to change the database authorization level. An encoded state of zero is returned by DBRC during the change authorization processing.
A9	Database Authorization error The command entered is not processed due to database authorization error.
AA	The `UPDATE DB` command is rejected because the database is in use by a BMP.
AB	The `UPDATE DB` command is rejected because the database is in use by a long-running DBCTL thread.
AC	The `UPDATE DB` command is rejected because the area of the DEDB is in long-busy wait.
AD	The `UPDATE DB` command is not successful due to a dynamic unallocation error.
AE	The `UPDATE DB` command is not successful due to a dynamic allocation error. No SVC99 is issued.
C1	Unknown DMB referenced for database The command cannot be processed because an unknown data management block is referenced for the database. Refer to the DFS564I message put out to the system console to identify the DMB name that cannot be referenced.
CC	OLR is active for database The command failed as OLR is active for the database.
D0	Database close error The command processing failed due to a database close error.
D1	Database write error The command processing failed due to a database write error.

Table 179. Completion Codes for UPDATE DB (continued)

Completion Code	Meaning
D2	Database needs backout The command processing failed as the database needs backout.
D3	Database or AREA needs recovery The command processing failed as the database or area needs recovery.
D4	Database needs image copy The command processing failed as the database needs image copy.
D5	Database has no backouts The command processing failed as there are no backouts for the database.
D6	Database in use A SET(ACCTYPE) is specified for the DEDB and the authorization level cannot be changed as the DEDB is in use in a region.
D7	Database I/O prevention not complete The database cannot be started as it is extended due to a XRF takeover and the I/O prevention is not complete.
D8	Database backouts pending The access type specified for the database cannot be changed as restartable backouts are pending for the database.
D9	Database open failed The command failed due to an error opening the database. Refer to DFS0730I messages to determine the reason of the failure.
DA	Database being recalled from HSM The command processing failed because the database is being recalled from HSM.
DB	Partition open failed The partition open failed because the master is offline.
DC	Database partition build failure The database partition build for the DDIR or DMB failed. Refer to the DFS0415I message sent to the system console to determine the reason of the failure.
DD	Database partition initialization failed The database partition initialization for the DDIR or DMB failed. Refer to the DFS0415 message sent to the system console for the details.
DE	ACBLIB read failure The command is not processed because there was an error reading the ACBLIB.
DF	Database directory initialization failed The command is not processed due to a database directory initialization failure.
E0	Database or area in recovery The command is not processed because the database or area is in recovery.
E2	Restart parallel DB open not complete The command is not processed because the restart parallel DB open is not complete for the database.
EE	Database backout error The command processing failed due to database backout error.

Table 179. Completion Codes for UPDATE DB (continued)

Completion Code	Meaning
EF	Database is in error The command entered is not processed because the database is in error.
F0	No AREA lock The command processing failed as the area lock could not be obtained.
F1	Area not stopped The command entered is not processed because the AREA is not stopped.
F2	Preload is active for AREA The command entered is not processed because the AREA is not stopped.
F3	Unresolved indoubts for AREA The command entered is not processed because unresolved indoubts exist for the AREA.
F4	Allocation failed Allocation failed for the AREA name.
F5	AREA needs recovery The command processing failed because the area needs recovery.
F6	ADS number discrepancy The command failed for the AREA because there is a discrepancy between the number of ADS allocated by IMS and the number of ADS known to DBRC. Correct the discrepancy and reissue the command to deallocate the datasets.
F7	Area is not loaded into CF The AREA OPEN failed and is not loaded into the Coupling Facility.
F8	Area has I/O tolerated CI The command is not processed because the AREA has an I/O tolerated CI.
F9	Area has 2nd CI EEQE The command is not processed because the AREA has a 2nd CI EEQE.

UPDATE DB Compared to Other Commands

Table 180 shows different instances of the `UPDATE DB` command and other IMS commands that perform similar functions.

Table 180. UPDATE DB Command Compared to Other IMS Commands

UPDATE DB Command	Similar IMS Commands
`UPDATE DB NAME(name) STOP(UPDATES)`	`/DBD DB dbname`
`UPDATE DB NAME(name) STOP(ACCESS)`	`/DBR DB dbname`
`UPDATE DB NAME(name) START(ACCESS)`	`/START DB dbname`
`UPDATE DB NAME(name) STOP(SCHD)`	`/STOP DB dbname`
`UPDATE DB NAME(name) SET(LOCK(ON))`	`/LOCK DB dbname`
`UPDATE DB NAME(name) SET(LOCK(OFF))`	`/UNLOCK DB dbname`

Examples for UPDATE DB Command

The following are examples of the UPDATE DB command:

Example 1 for UPDATE DB Command

TSO SPOC input:

```
UPD DB NAME(BE3PARTS PDHDOKA D2XHDJ05 DEDBJN21) START(ACCESS)
```

TSO SPOC output:

```
DBName   MbrName    CC
BE3PARTS IMSA       A0
BE3PARTS IMS1       A0
DEDBJN21 IMSA        0
DEDBJN21 IMS1        0
D2XHDJ05 IMSA        0
D2XHDJ05 IMS1        0
PDHDOKA  IMSA        0
PDHDOKA  IMS1        0
```

OM API input:

```
CMD (UPD DB NAME(BE3PARTS PDHDOKA D2XHDJ05 DEDBJN21) START(ACCESS))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.2.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2003.128 21:41:33.429986</statime>
<stotime>2003.128 21:41:33.548653</stotime>
<staseq>B963E429B4CE2C4A</staseq>
<stoseq>B963E429D1C6D02E</stoseq>
<rqsttkn1>USRT005 10144133</rqsttkn1>
<rc>0200000C</rc>
<rsn>00003008</rsn>
</ctl>
<cmderr>
<mbr name="IMSA    ">
<typ>IMS     </typ>
<styp>DBCTL   </styp>
<rc>0000000C</rc>
<rsn>00003000</rsn>
</mbr>
<mbr name="IMS1    ">
<typ>IMS     </typ>
<styp>DBDC    </styp>
<rc>0000000C</rc>
<rsn>00003000</rsn>
</mbr>
</cmderr>
<cmd>
<master>IMSA    </master>
<userid>USRT005 </userid>
<verb>UPD </verb>
<kwd>DB              </kwd>
<input>UPD DB NAME(BE3PARTS PDHDOKA D2XHDJ05 DEDBJN21) START(ACCESS)</input>
</cmd>
<cmdrsphdr>
<hdr slbl="DB" llbl="DBName" scope="LCL" sort="a" key="1" scroll="no"
 len="8" dtype="CHAR" align="left" skipb="no"  />
<hdr slbl="AREA" llbl="AreaName" scope="LCL" sort="a" key="3"
 scroll="no" len="8" dtype="CHAR" align="left" skipb="yes" />
<hdr slbl="MBR" llbl="MbrName" scope="LCL" sort="a" key="2" scroll="no"
 len="8" dtype="CHAR" align="left" skipb="no" />
```

```
<hdr slbl="CC" llbl="CC" scope="LCL" sort="n" key="0" scroll="yes"
 len="4" dtype="INT" align="right" skipb="no" />
<hdr slbl="ERRT" llbl="ErrorText" scope="LCL" sort="n" key="0"
 scroll="yes" len="16" dtype="CHAR" align="left" skipb="yes" />
</cmdrsphdr>
<cmdrspdata>
<rsp>DB(DEDBJN21) MBR(IMSA    ) CC(   0) </rsp>
<rsp>DB(BE3PARTS) MBR(IMSA    ) CC(  A0) </rsp>
<rsp>DB(D2XHDJ05) MBR(IMSA    ) CC(   0) </rsp>
<rsp>DB(PDHDOKA ) MBR(IMSA    ) CC(   0) </rsp>
<rsp>DB(DEDBJN21) MBR(IMS1    ) CC(   0) </rsp>
<rsp>DB(BE3PARTS) MBR(IMS1    ) CC(  A0) </rsp>
<rsp>DB(D2XHDJ05) MBR(IMS1    ) CC(   0) </rsp>
<rsp>DB(PDHDOKA ) MBR(IMS1    ) CC(   0) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: The command starts access of the listed databases and makes them available. Any error in starting the database is returned as a completion code.

Example 2 for UPDATE DB Command

TSO SPOC input:

```
UPD DB NAME(BE3PARTS DEDBJN03 PVHDJ5A DBHDOK01 ) STOP(ACCESS)
```

TSO SPOC output:

```
DBName   AreaName MbrName    CC
BE3PARTS          IMSA        0
BE3PARTS          IMS1        0
DBHDOK01          IMSA        0
DBHDOK01          IMS1        0
DEDBJN03          IMSA        0
DEDBJN03          IMS1        C
DEDBJN03 DB3AREA0 IMS1       F0
PVHDJ5A           IMSA        0
PVHDJ5A           IMS1        0
```

OM API input:

```
CMD(UPD DB NAME(BE3PARTS DEDBJN03 PVHDJ5A DBHDOK01 ) STOP(ACCESS))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.2.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2003.129 22:41:53.466082</statime>
<stotime>2003.129 22:41:53.680616</stotime>
<staseq>B965338380AE2E8E</staseq>
<stoseq>B9653383B50E8663</stoseq>
<rqsttkn1>USRT005 10154153</rqsttkn1>
<rc>0200000C</rc>
<rsn>00003000</rsn>
</ctl>
<cmderr>
<mbr name="IMS1    ">
<typ>IMS     </typ>
<styp>DBDC    </styp>
<rc>0000000C</rc>
<rsn>00003000</rsn>
</mbr>
</cmderr>
<cmd>
<master>IMSA    </master>
```

```
<userid>USRT005 </userid>
<verb>UPD </verb>
<kwd>DB              </kwd>
<input>UPD DB NAME(BE3PARTS DEDBJN03 PVHDJ5A DBHDOK01 ) STOP(ACCESS)     </input>
</cmd>
<cmdrsphdr>
<hdr slbl="DB" llbl="DBName" scope="LCL" sort="a" key="1" scroll="no"
 len="8" dtype="CHAR" align="left" skipb="no"  />
<hdr slbl="AREA" llbl="AreaName" scope="LCL" sort="a" key="3"
 scroll="no" len="8" dtype="CHAR" align="left" skipb="yes" />
<hdr slbl="MBR" llbl="MbrName" scope="LCL" sort="a" key="2" scroll="no"
 len="8" dtype="CHAR" align="left" skipb="no" />
<hdr slbl="CC" llbl="CC" scope="LCL" sort="n" key="0" scroll="yes"
 len="4" dtype="INT" align="right" skipb="no" />
<hdr slbl="ERRT" llbl="ErrorText" scope="LCL" sort="n" key="0"
 scroll="yes" len="16" dtype="CHAR" align="left" skipb="yes" />
</cmdrsphdr>
<cmdrspdata>
<rsp>DB(DEDBJN03) MBR(IMSA    ) CC(   0) </rsp>
<rsp>DB(BE3PARTS) MBR(IMSA    ) CC(   0) </rsp>
<rsp>DB(DBHDOK01) MBR(IMSA    ) CC(   0) </rsp>
<rsp>DB(PVHDJ5A ) MBR(IMSA    ) CC(   0) </rsp>
<rsp>DB(DEDBJN03) AREA(DB3AREA0) MBR(IMS1    ) CC(  F0) </rsp>
<rsp>DB(DEDBJN03) MBR(IMS1    ) CC(   C) </rsp>
<rsp>DB(BE3PARTS) MBR(IMS1    ) CC(   0) </rsp>
<rsp>DB(DBHDOK01) MBR(IMS1    ) CC(   0) </rsp>
<rsp>DB(PVHDJ5A ) MBR(IMS1    ) CC(   0) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: The `UPD DB NAME(BE3PARTS DEDBJN03 PVHDJ5A DBHDOK01) STOP(ACCESS)` command stops access to the databases BE3PARTS, DEDBJN03, PVHDJ5A, and DBHDOK01 and takes them offline. Response lines are returned for all the databases that are processed. Response lines are also returned for the DEDB areas that resulted in an error.

Example 3 for UPDATE DB Command

TSO SPOC input:

```
UPD DB NAME(*) STOP(SCHD)
```

TSO SPOC output:

```
Log for . . : UPD DB NAME(*) STOP(SCHD)
IMSplex . . . . . : PLEX1
Routing . . . . . :
Start time. . . . : 2003.128 15:06:15.89
Stop time . . . . : 2003.128 15:06:15.92
Return code . . . : 00000000
Reason code . . . : 00000000
Command master. . : IMSA
```

OM API input:

```
CMD(UPD DB NAME(*) STOP(SCHD))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.2.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2003.128 22:06:15.898464</statime>
<stotime>2003.128 22:06:15.916720</stotime>
<staseq>B963E9AF7F96026A</staseq>
<stoseq>B963E9AF840B0E06</stoseq>
```

```
<rqsttkn1>USRT005 10150615</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<master>IMSA    </master>
<userid>USRT005 </userid>
<verb>UPD </verb>
<kwd>DB              </kwd>
<input>UPD DB NAME(*) STOP(SCHD) </input>
</cmd>
</imsout>
```

Explanation: The UPD DB NAME(*) STOP(SCHD) command is routed to all IMSs to stop the scheduling of all the databases and to prevent subsequently scheduled programs from accessing them. The command is successful for all IMSs. No response lines are returned because the default action for NAME(*) is to return response lines only for area names that resulted in an error. OPTION(ALLRSP) can be specified to obtain all the database names processed by the command.

UPDATE LE

The UPDATE LE command allows the user to define Language Environment (LE) runtime parameter overrides or to change the system option to enable/disable LE override processing. The parameters can be filtered by a transaction code, LTERM name, userid, or program name for MPP and JMP regions. The parameters may be filtered by a program name for IFB, BMP, and JBP regions. Message driven BMP regions can also filter on a transaction code. Any combination of parameters may be used to qualify the application instance to which the runtime parameters are applied. The first available entry in the table is used. The new entry may be added before or after existing entries, depending on where free space exists in the table.

This command may be specified only through the Operations Manager API. The command syntax for this command is defined in XML and is available to automation programs that communicate with OM.

OM overrides the routing on the command and routes the command to all IMS systems in the IMSplex. The user specified route list is ignored.

At least one of the resource filters--TRAN, LTERM, USERID, or PGM--must be specified.

LTERM()
Specifies the 1-8 character name of the lterm to which the parameters are to be defined.

PGM() Specifies the 1-8 character name of the program to which the parameters are to be defined.

SET() Specifies the updates to attributes or parameters.

LEOPT()
Specifies whether or not LE runtime parameters can be overridden dynamically for all active IMS systems in the IMSplex. YES indicates that the IMS systems allow overrides which enables the DL/I INQY call to retrieve runtime parameters. NO indicates that the IMS systems do not allow dynamic overrides to the parameters.

This option allows the user to override the LEOPT=Y|N option that is specified in the DFSCGxxx PROCLIB member without having to bring down the IMS system.

When runtime overrides are disabled (IMS is started with LEOPT=N or the UPD LE SET(LEOPT(NO)) command is issued) the runtime parameter table continues to be updated as `UPD LE SET(LERUNOPTS())` or `DEL LE` commands are issued. If the `UPD LE SET(LEOPT(YES))` command is then specified, all changes that were made during the time overrides were disabled are available.

This command is automatically routed to all IMS systems that are active in the IMSPlex. The user can not override the OM routing to route to a single IMS.

LERUNOPTS()
Specifies the LE dynamic runtime parameters. If an existing set of parameters is found for the specified TRAN, LTERM, USERID, or PGM, the new parameter string completely replaces the existing parameter string. The parameters are not appended to the existing string. The string is only replaced when the specified filters are an exact match for the existing entry.

If there is no existing entry, then the first unused entry that is large enough to contain the parameters is used for the update. An unused entry is one that was previously deleted by the `DEL LE` command. If there are no unused entries or none that are large enough, storage is allocated for a new entry and the entry is added to the top of the table.

The `UPDATE LE SET(LERUNOPTS())` command is processed regardless of the LEOPT system option. This means that IMS continues to build the runtime parameter table entries even though they will not be retrieved through the DL/I INQY call.

This command is automatically routed to all IMS systems that are active in the IMSPlex. The user cannot override the parameters on a single IMS.

TRAN()
Specifies the 1-8 character name of the transaction to which the parameters are to be defined.

USERID()
Specifies the 1-8 character name of the userid to which the parameters are to be defined.

UPDATE LE Output Fields

Table 181 on page 740 shows the `UPDATE LE` output fields. The columns in the table are as follows:

Short Label
Contains the short label generated in the XML output.

Keyword
Identifies the keyword on the command that caused the field to be generated. N/A appears for output fields that are always returned. ERR appears for output fields that are returned only in case of an error.

Meaning
Provides a brief description of the output field.

Table 181. Output Fields for the UPDATE LE Command

Short Label	Keyword	Meaning
CC	N/A	Completion code for the line of output. Completion code is always returned.
MBR	N/A	IMSplex member (IMS identifier) that built the output line. Member name is always returned.

Return, Reason, and Completion Codes for UPDATE LE

The OM return and reason codes that may be returned as a result of this command are standard for all commands entered through the OM API. An IMS return and reason code is returned to OM by the UPDATE LE command. In Table 182 the return and reason code for the UPDATE LE command are displayed.

Table 182. Return and Reason Codes for the UPDATE LE Command

Return Code	Reason Code	Meaning
X'00000000'	X'00000000'	The UPDATE LE command completed successfully.
X'00000008'	X'00002008'	No keywords were specified on the command. At least one keyword is required. When SET(LERUNOPTS()) is specified, at least one resource name must be specified.
X'00000008'	X'00002010'	An asterisk or percentage sign was specified in the filter name. Wildcards are not supported on the command.
X'00000008'	X'00002014'	An invalid character was specified in the filter name.
X'00000008'	X'00002040'	An invalid filter was specified on the command. When SET(LEOPT()) is specified, no other resource names can be specified.
X'00000010'	X'00004040'	The parameter override header has not been initialized. Retry the command after restart is complete.
X'00000014'	X'00005000'	Unable to get storage from IMODULE GETSTOR.
X'00000014'	X'00005010'	Unable to obtain latch.
X'00000014'	X'00005FFF'	Internal IMS error - should not occur.

Table 183 contains the completion code that can be returned on a UPDATE LE command.

Table 183. Completion Code for the UPDATE LE Command

Completion Code	Meaning
0	The UPDATE LE command completed successfully for the specified resource.

Examples for UPDATE LE Command

The following are examples of the UPDATE LE command.

Example 1 for UPDATE LE Command

Assume the following filters and parameters are specified on UPD LE SET(LERUNOPTS()) commands that are processed in the order listed.

1. TRAN(PART) LTERM(TERM2) SET(LERUNOPTS(hhhh))

2. TRAN(PART) LTERM(TERM2) SET(LERUNOPTS(iiii))
3. LTERM(TERM2) USERID(BETTY) SET(LERUNOPTS(gggg))
4. TRAN(PART) LTERM(TERM1) USERID(BOB) SET(LERUNOPTS(ffff))
5. TRAN(PART) LTERM(TERM1) USERID(BARBARA) SET(LERUNOPTS(eeee))
6. PGM(DFSSAM02) SET(LERUNOPTS(dddd))
7. TRAN(PART) LTERM(TERM1) SET(LERUNOPTS(cccc))
8. TRAN(PART) USERID(BETTY) SET(LERUNOPTS(bbbb))
9. TRAN(PART) PGM(DFSSAM02) SET(LERUNOPTS(aaaa))

Rules for matching an entry which results in an update of an existing entry:

- The number of filters defined on the `UPDATE LE` must match the number of filters defined in the entry.
- The filter values defined on the `UPDATE LE` must be an exact match for those defined in the entry.

Table 184 is a logical representation of the parameter override table entries at the end of the command processing. The table includes the transaction name, LTERM, USERID, Program, and LERUNOPTS for each entry.

Table 184. Parameter Override Table Entries for UPDATE LE Example 1

Entry#	TRAN	LTERM	USERID	PROGRAM	LERUNOPTS
1	PART			DFSSAM02	aaaa
2	PART		BETTY		bbbb
3	PART	TERM1			cccc
4				DFSSAM02	dddd
5	PART	TERM1	BARBARA		eeee
6	PART	TERM1	BOB		ffff
7		TERM2	BETTY		gggg
8	PART	TERM2			iiii

Example 2 for UPDATE LE Command

TSO SPOC input:

```
UPD LE TRAN(IAPMDI26) USERID(USRT001)
SET(LERUNOPTS(RPTOPTS=((ON),NOOVR),RPTSTG=((OFF),NOOVR)))
```

TSO SPOC output:

```
MbrName    CC
SYS3        0
```

OM API input:

```
CMD(UPD LE TRAN(IAPMDI26) USERID(USRT001)
SET(LERUNOPTS(RPTOPTS=((ON),NOOVR),RPTSTG=((OFF),NOOVR))))
```

OM API output:

```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2002.163 17:56:10.220516</statime>
<stotime>2002.163 17:56:10.221547</stotime>
```

```
<staseq>B7C4CA4EDBFE420E</staseq>
<stoseq>B7C4CA4EDC3EB382</stoseq>
<rqsttkn1>USRT002 10105610</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<master>SYS3    </master>
<userid>USRT002 </userid>
<verb>UPD </verb>
<kwd>LE              </kwd>
<input>UPD LE TRAN(IAPMDI26) USERID(USRT001)
 SET(LERUNOPTS(RPTOPTS=((ON),NOOVR),RPTSTG=((OFF),NOOVR)))
</input>
</cmd>
<cmdrsphdr>
<hdr slbl="MBR" llbl="MbrName" scope="LCL" sort="A" key="1" scroll="NO"  len="8"
dtype="CHAR" align="left" />
<hdr slbl="CC"  llbl="CC"      scope="LCL" sort="N" key="0" scroll="YES" len="4"
dtype="INT" align="right" />
</cmdrsphdr>
<cmdrspdata>
<rsp>MBR(SYS3    ) CC(   0) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: The UPDATE LE command adds an entry to the LE runtime options table. The entry added by this command defines two filters: userid and transaction. The transaction is set to IAPMDI26 and the userid is set to USRT001. The runtime options string for this table entry is RPTOPTS=((ON),NOOVR),RPTSTG=((OFF),NOOVR). The output shows that IMS member SYS3 processed the command with a return code of 0.

UPDATE TRAN

The UPDATE TRAN command changes the status of transaction resources in an IMSplex.

This command may be specified only through the OM API.

The UPDATE TRAN command is not valid on the RSR tracker.

The UPDATE TRAN SET(MAXRGN) and UPDATE TRAN STARTISTOP(TRACE) commands are valid on the XRF alternate. All other instances of the UPDATE TRAN command are not valid on the XRF alternate.

The UPDATE TRAN command with START(Q,SCHD,SUSPEND), STOP(Q,SCHD), or SET(CLASS,CPRI,LCT,LPRI,MAXRGN,NPRI,PARLIM,PLCT,SEGNO,SEGSZ) is recoverable over an IMS restart. The UPDATE TRAN command with START(TRACE), or STOP(TRACE) is not recoverable over an IMS restart.

A new X'22' log record is written during the processing of the UPDATE TRAN command with START(Q,SCHD,SUSPEND), STOP(Q,SCHD), or SET(CLASS,CPRI,LCT,LPRI,MAXRGN,NPRI,PARLIM,PLCT,SEGNO,SEGSZ). A log record is not written for the UPDATE TRAN command with START(TRACE) or STOP(TRACE).

The command syntax for this command is defined in XML and is available to automation programs that communicate with OM.

CLASS()
Selects the transactions associated with the specified class or classes to be updated.

NAME()
Specifies the names of the specific transactions that are to be updated or a group of transactions whose names match a generic wild card parameter.

SCOPE()
Specifies where IMS should apply the change. The default is ALL. When ALL is specified, the changes are applied to the active IMS systems. When ACTIVE is specified, the changes are applied to the IMS systems currently active. ACTIVE and ALL produce the same result.

SET() Specifies the attribute values to be changed.

CLASS
Specifies a new processing class for the transaction. The new class takes effect the next time the transaction is scheduled. Valid CLASS parameters are numeric values from 1 to 999.

CPRI Specifies a new value for the current priority of a transaction. The CPRI keyword is not allowed for BMP transactions, because BMP transactions should always have a priority of 0. The new CPRI value takes effect the next time the transaction is scheduled. Valid CPRI parameters are numeric values from 0 to 14.

LCT Specifies a new value for the limit count of a transaction. The new LCT value takes effect during execution of the transaction. Valid LCT parameters are numeric values from 1 to 65535.

LPRI Specifies a new value for the limit priority of a transaction. The LPRI keyword is not allowed for BMP transactions, because BMP transactions should always have a priority of 0. The new LPRI value takes effect the next time the transaction is scheduled. Valid LPRI parameters are numeric values from 0 to 14.

MAXRGN
Specifies a new value for the maximum number of regions that can be simultaneously scheduled for a given transaction. The transaction must be eligible for parallel scheduling (load balancing). The value of the MAXRGN parameter must be between 0 and the number specified on the MAXPST=control region parameter.

NPRI Specifies a new value for the normal priority of a transaction. The NPRI keyword is not allowed for BMP transactions, because BMP transactions should always have a priority of 0. The new NPRI value takes effect the next time the transaction is scheduled. Valid NPRI parameters are numeric values from 0 to 14.

PARLIM
Specifies a new value for the parallel processing limit count of a transaction. PARLIM is the maximum number of messages that can currently be queued, but not yet processed, by each active message region currently scheduled for this transaction. An additional region is scheduled whenever the transaction queue count exceeds the PARLIM value multiplied by the number of regions currently scheduled for this transaction. Valid PARLIM parameters are numeric values from 0 to 32767 and 65535, where 65535 disables transaction load balancing.

PARLIM can not be specified for CPI Communications driven transactions.

PLCT Specifies a new value for the processing limit count of a transaction. The PLCT is the number of messages of this transaction code that a program can process in a single scheduling. The new PLCT values take effect the next time the transaction is scheduled. Valid PLCT parameters are numeric values from 0 to 65535.

PLCT can not be specified for CPI Communications driven transactions.

SEGNO
Specifies a new value for the limit on the number of application program output segments allowed in message queues for each GU call. Segment limits are established by transaction code, thereby allowing specification of more than one value for each application program. The new SEGNO value takes effect during execution of the transaction. Valid SEGNO parameters are numeric values from 0 to 65535.

SEGNO can not be specified for CPI Communications driven transactions.

SEGSZ
Specifies a new value for the size of application program output segments allowed in message queues for each GU call. The new SEGSZ value takes effect during execution of the transaction. Valid SEGSIZE parameters are numeric values from 0 to 65535.

SEGSZ can not be specified for CPI Communications driven transactions.

START()
Specifies the attributes to be started.

Q Starts the queueing of messages.

SCHD Starts the scheduling of messages.

SUSPEND
If the transaction has messages on the suspend queue, that suspend queue is automatically transferred to the ready queue.

TRACE
Starts the transaction trace, which captures the DL/I portion of Data Communications (DC) for the specified transaction. The information is written as a 6701 log record to the IMS log. For more information about this trace, see the discussion of the DLA3LOG trace in *IMS Version 9: Diagnosis Guide and Reference*

STOP()
Specifies the attributes to be stopped.

Q Stops the queueing of messages. However, output can still be queued if it originates from the application program.

SCHD Stops the scheduling of messages.

TRACE
Stops the transaction trace.

UPDATE TRAN Output Fields

Table 185 shows the UPDATE TRAN output fields. The columns in the table are as follows:

Short Label
Contains the short label generated in the XML output.

Keyword
Identifies the keyword on the command that caused the field to be generated. N/A appears for output fields that are always returned. ERR appears for output fields that are returned only in case of an error.

Meaning
Provides a brief description of the output field.

Table 185. Output Fields for the UPDATE TRAN Command

Short Label	Keyword	Meaning
CC	N/A	Completion code for the line of output. The completion code is always returned.
MBR	N/A	The IMSplex member that built the output line. The IMS identifier of the IMS for which the transaction information is displayed. The IMS identifier is always returned.
TRAN	N/A	The transaction name. The transaction name is always displayed.

Return, Reason, and Completion Codes for UPDATE TRAN

Table 186 contains the return and reason codes for the UPDATE TRAN command.

Table 186. Return and Reason Codes for the UPDATE TRAN Command

Return Code	Reason Code	Meaning
X'00000000'	X'00000000'	The UPDATE TRAN command completed successfully.
X'00000004'	X'00001000'	The UPDATE TRAN command was not processed on the IMS system as the IMS system is not the command master. No resource information is returned.
X'00000008'	X'0000200C'‘	No resources were found to be updated. The resource name(s) specified may be invalid or there were no resources that match the filter specified. Confirm that the UPDATE TRAN command is issued with valid resources.
X'00000008'	X'00002040'	More than one filter value is specified on the UPDATE TRAN command. Confirm that only one of SET or START | STOP filters are specified on the command.
X'00000008'	X'00002044'	The UPDATE TRAN command is not processed because the same attribute value was specified for the START and STOP filters. The attribute ″Q,SCHD″ can be specified only on START or STOP but not both. For example, UPDATE TRAN START(Q) STOP(Q) is not valid but UPDATE TRAN START(Q) STOP(SCHD) is valid. Confirm that only one START | STOP attribute is specified on the command.
X'00000008'	X'00002048'	The UPDATE TRAN command is not processed because an invalid SET attribute is specified. Confirm that the correct SET attribute is specified on the command.

Table 186. Return and Reason Codes for the UPDATE TRAN Command (continued)

Return Code	Reason Code	Meaning
X'00000008'	X'0000204C'	The UPDATE TRAN command is not processed because a CLASS value specified is invalid. Confirm that the correct CLASS value is specified on the command.
X'00000008'	X'00002050'	The UPDATE TRAN command is not processed because the CPRI value specified is invalid. Confirm that the correct CPRI value is specified on the command.
X'00000008'	X'00002054'	The UPDATE TRAN command is not processed because the LCT (limit count) value specified is invalid. Confirm that the correct LCT value is specified on the command.
X'00000008'	X'00002058'	The UPDATE TRAN command is not processed because the LPRI value specified is invalid. Confirm that the correct LPRI value is specified on the command.
X'00000008'	X'0000205C'	The UPDATE TRAN command is not processed because the MAXGN value specified is invalid. Confirm the correct MAXRGN value is specified on the command.
X'00000008'	X'00002060'	The UPDATE TRAN command is not processed because the NPRI value specified is invalid. Confirm that the correct NPRI value is specified on the command.
X'00000008'	X'00002064'	The UPDATE TRAN command is not processed because the PARLIM value is invalid. Confirm that the PARLIM value is specified on the command.
X'00000008'	X'00002068'	The UPDATE TRAN command is not processed because the PLCT value is invalid. Confirm that the correct PLCT value is specified on the command.
X'00000008'	X'0000206C'	The UPDATE TRAN command is not processed because the SEGNO value specified is invalid. Confirm that the correct SEGNO value is specified on the command.
X'00000008'	X'00002070'	The UPDATE TRAN command is not processed because the SEGSZ value specified is invalid. Confirm that the correct SEGSZ value is specified on the command.
X'0000000C'	X'00003000'	The UPDATE TRAN command was successful for at least one resource name. The UPDATE TRAN command was not successful for one or more resource names. The completion code indicates the reason for the error with the resource name. The completion codes that can be returned by the UPDATE TRAN command are listed in the UPDATE TRAN Completion Code table.
X'0000000C'	X'00003004'	The UPDATE TRAN command was not successful for all the resource names specified. The completion code indicates the reason for the error with the resource name. The completion code that can be returned by the UPDATE TRAN command is listed in the UPDATE TRAN Completion Code table.
X'00000010'	X'0000400C'	The UPDATE TRAN command is not processed as the flavor entered is not valid on the XRF alternate.
X'00000010'	X'00004014'	The UPDATE TRAN command is not processed as the TRAN keyword is not valid on the RSR tracker.
X'00000014'	X'00005004'	The UPDATE TRAN command processing terminated because a DFSOCMD response buffer could not be obtained.

Table 186. Return and Reason Codes for the UPDATE TRAN Command (continued)

Return Code	Reason Code	Meaning
X'00000014'	X'00005008'	The UPDATE TRAN command processing terminated as DFSPOOL storage could not be obtained.
X'00000014'	X'0000500C'	The UPDATE TRAN command processing terminated an AWE could not be obtained.

Errors unique to the processing of the UPDATE TRAN command are returned as a completion code. A completion code is returned for each action against an individual resource. The following completion codes in Table 187 may be returned on an UPDATE TRAN command.

Table 187. Completion Code for the UPDATE TRAN Command

Completion Code	Meaning
0	The UPDATE TRAN command completed successfully for the resource.
10	The resource name is unknown to the client that is processing the request. The resource name may have been typed in error or the resource may not be active at this time. If this is a wildcard request there were no matches for the name. Confirm that the correct spelling of the resource name is specified on the command.
30	The UPDATE TRAN command is invalid for the resource because the transaction is a dynamic transaction that was created during shared queues processing.
34	The UPDATE TRAN command is invalid for the resource because the transaction is a CPIC transaction.
38	The UPDATE TRAN command is invalid for the resource because the PSB associated with the transaction is a BMP.
3C	The MAXRGN can not be changed for the resource because the transaction does not have parallel limits specified.
40	The PARLIM can not be changed for the resource because the PSB associated with the transaction is defined as does not have parallel scheduling.
44	The UPDATEUPDATE TRAN command can not be processed for the resource because the transaction is currently being scheduled.
45	The UPDATE TRAN command could not be completed for the resource because the SID number is invalid.

UPDATE TRAN Compared to Other Commands

Table 188 shows different instances of the UPDATE TRAN command and other IMS commands that perform similar functions.

Table 188. UPDATE TRAN Compared to Other Commands

UPDATE Command	Similar IMS Command
UPDATE TRAN(name) START(Q) STOP(SCHD)	/PSTOP TRAN name
UPDATE TRAN NAME(name) START(SCHD) STOP(Q)	/PUR TRAN name
UPDATE TRAN NAME(name) START(Q,SCHD,SUSPEND)	UPDATE TRAN NAME(name)
UPDATE TRAN NAME(name) STOP(Q,SCHD)	/STOP TRAN name

Table 188. UPDATE TRAN Compared to Other Commands (continued)

UPDATE Command	Similar IMS Command
UPDATE TRAN NAME(name) START(TRA)	/TRA SET ON TRAN name
UPDATE TRAN NAME(name) STOP(TRA)	/TRA SET OFF TRAN name
UPDATE TRAN NAME(name) SET(CLASS(new_class_number))	/ASSIGN TRAN name TO CLS new_class_number
UPDATE TRAN NAME(name) SET(CPRI(new_current_priority))	/ASSIGN CPRI new_current_priority TO TRAN name
UPDATE TRAN NAME(name) SET(LCT(new_limit_count))	/ASSIGN LCT new_lmct_number TO TRAN name
UPDATE TRAN NAME(name) SET(LPRI(new_limit_priority))	/ASSIGN LPRI new_lpri_number TO TRAN name
UPDATE TRAN NAME(name) SET(NPRI(new_normal_priority))	/ASSIGN NPRI new_npri_number TO TRAN name
UPDATE TRAN NAME(name) SET(PARLIM(new_parallel_limit))	/ASSIGN PARLIM new_parmlim_number TO TRAN name
UPDATE TRAN NAME(name) SET(PLCT(new_processing_limit))	/ASSIGN PLCT new_plmct_number TO TRAN name
UPDATE TRAN NAME(name) SET(SEGNO(new_segment_number))	/ASSIGN SEGNO new_segno_number TO TRAN name
UPDATE TRAN NAME(name) SET(SEGSZ(new_segment_size))	/ASSIGN SEGSZ new_segsize_number TO TRAN name
UPDATE TRAN NAME(name) SET(MAXRG(new_max_regions))	/CHA TRAN name MAXRGN new_maxrgn_number

Example 1 for UPDATE TRAN Command

TSO SPOC input:

```
UPDATE TRAN NAME(ADDINV) STOP(Q,SCHD)
```

TSO SPOC output:

```
Trancode MbrName    CC
ADDINV   SYS3        0
```

OM API input:

```
CMD(UPDATE TRAN NAME(ADDINV) STOP(Q,SCHD))
```

OM API output:

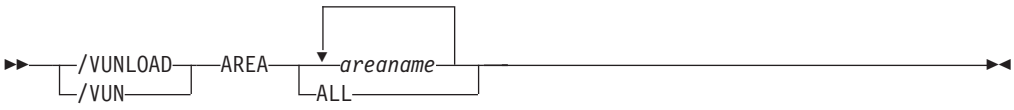
```
<imsout>
<ctl>
<omname>OM1OM   </omname>
<omvsn>1.1.0</omvsn>
<xmlvsn>1   </xmlvsn>
<statime>2002.163 17:11:29.656126</statime>
<stotime>2002.163 17:11:29.682014</stotime>
<staseq>B7C4C0527953EC28</staseq>
<stoseq>B7C4C0527FA5E144</stoseq>
<rqsttkn1>USRT002 10101129</rqsttkn1>
<rc>00000000</rc>
<rsn>00000000</rsn>
</ctl>
<cmd>
<master>SYS3    </master>
<userid>USRT002 </userid>
```

```
<verb>UPD </verb>
<kwd>TRAN            </kwd>
<input>UPDATE TRAN NAME(ADDINV) STOP(Q,SCHD)             </input>
</cmd>
<cmdrsphdr>
<hdr slbl="TRAN" llbl="Trancode" scope="LCL" sort="a" key="1" scroll="no" len="8"
dtype="CHAR" align="left" />
<hdr slbl="MBR"  llbl="MbrName"  scope="LCL" sort="a" key="2" scroll="no" len="8"
dtype="CHAR" align="left" />
<hdr slbl="CC"   llbl="CC"       scope="LCL" sort="n" key="0" scroll="yes" len="4"
dtype="INT" align="right" /></cmdrsphdr>
<cmdrspdata>
<rsp>TRAN(ADDINV  ) MBR(SYS3    ) CC(   0) </rsp>
</cmdrspdata>
</imsout>
```

Explanation: Queuing and scheduling is stopped for transaction ADDINV.

Chapter 66. /VUNLOAD

Format



Environments and Keywords

Table 189 lists the environments (DB/DC, DBCTL, and DCCTL) from which the command and keyword can be issued.

Table 189. Valid Environments for the /VUNLOAD Command and Keyword

COMMAND / KEYWORD	DB/TM	DBCTL	DCCTL
/VUNLOAD	X	X	
AREA	X	X	

Usage

The /VUNLOAD AREA command removes the specified areas from the z/OS data space or coupling facility. All of the updated CIs for the area are written to DASD. All subsequent I/O for the area is from DASD. /VUNLOAD processing occurs concurrently with application processing.

In a data-sharing environment, in order to maintain data integrity, IMS requests IRLM to lock the first control interval (CI) in exclusive mode before IMS unloads the area from the coupling facility. This lock causes those IMS subsystems sharing the area to complete any synchronization point processing. When all sharing subsystems are no longer using the area, IMS requests IRLM to release the CI locks and IMS unloads the area. All subsequent I/O for the area is from DASD.

The /VUNLOAD command does not change any of the VSO options set in the RECON data set. Therefore, at the next IMS restart or /START AREA command, the VSO options again take effect. Any changes to VSO definitions must be made through DBRC commands. For Shared VSO areas however, the /VUNLOAD command is persistent across IMS restarts and can be reset only by a /START AREA command. This is true for shared VSO areas for consistency across all sharing partners. This prevents one system from accessing data from DASD while others are accessing from the coupling facility.

For multiple VSO DEDB areas that share a single coupling facility structure, when the /VUNLOAD AREA command is issued, one of the following actions may occur:

- If there is only one area in the structure (either a single-area structure or a multi-area structure with one area), the area is taken out of VSO and the structure is deleted. The IMS issuing the /VUNLOAD AREA command notifies the other IMSs sharing the area to disconnect from the structure.
- If there are multiple areas in the structure, when one area is taken out of VSO, the area is disassociated and its CIs are deleted in the structure. The IMS issuing the /VUNLOAD AREA notifies the other IMSs sharing the area to either

© Copyright IBM Corp. 1974, 2003

disassociate or disconnect from the structure depending on if the IMS is still sharing or not sharing other areas in the structure.

The /VUNLOAD command is not valid for an RSR tracking subsystem.

Example for /VUNLOAD Command

Entry ET:

```
/VUNLOAD
```

Response ET:

```
NC0000000 FPEC 03070 09:58:03.52 01   00000290  R 14,/VUN AREA DD01AR0
NR8400000 FPEC 03070 09:58:03.52 JOB00116 00000090  IEE600I REPLY TO 14 IS;/VUN AREA DD01AR0
N 8400000 FPEC 03070 09:58:03.53 JOB00116 00000090  DFS058I 09:58:03 VUNLOAD COMMAND IN PROGRESS   SYS3
W 8400000 FPEC 03070 09:58:03.53 JOB00116 00000090 *17 DFS996I *IMS READY*  SYS3
N 8400000 FPEC 03070 09:58:04.42 JOB00116 00000090  DFS2823I AREA DD01AR0  DISCONNECT FROM STR: DD01AR0STR1  SUCCE
N 8400000 FPEC 03070 09:58:04.42 JOB00116 00000090  DFS0488I VUN COMMAND COMPLETED. AREA= DD01AR0  RC= 0 SYS3
```

Part 3. z/OS (MVS) Commands Used for IMS

Chapter 67. Introduction . . . 755

Chapter 68. START FDBRPROC . . . 757
Format . . . 757
Usage. . . . 757

Chapter 69. MODIFY IMS . . . 759
F jobname,DUMP . . . 759
Format . . . 759
Usage. . . . 759
Example . . . 759
F jobname,DUMPxxxx . . . 759
Format . . . 759
Usage. . . . 759
Examples . . . 760
F jobname,FORCExxxx . . . 760
Format . . . 760
Usage. . . . 760
Examples . . . 760
F jobname,RECONNECT . . . 761
Format . . . 761
Usage. . . . 761
Example . . . 761
F jobname,RECONNSTR . . . 762
Format . . . 762
Usage. . . . 762
Example . . . 762
F jobname,STOP . . . 763
Format . . . 763
Usage. . . . 763
Example . . . 763
F jobname,STOPxxxx . . . 763
Format . . . 763
Usage. . . . 763
Examples . . . 763

Chapter 70. START IRLMPROC . . . 765
Format . . . 765
Usage. . . . 765

Chapter 71. MODIFY FDBRPROC . . . 769
F fdbrproc,DUMP . . . 769
Format . . . 769
Usage. . . . 769
F fdbrproc,RECOVER . . . 769
Format . . . 769
Usage. . . . 769
F fdbrproc,STATUS . . . 769
Format . . . 769
Usage. . . . 769
F fdbrproc,STOP . . . 770
Format . . . 770
Usage. . . . 771

© Copyright IBM Corp. 1974, 2003

F fdbrproc,TERM 771
Format 771
Usage. 771

Chapter 72. MODIFY IRLMPROC 773
F irlmproc,ABEND 773
Format 773
Usage. 773
Examples 773
F irlmproc,DIAG,HANG 774
Format 774
Usage. 774
F irlmproc,PURGE,imsname 775
Format 775
Usage. 775
Example 775
F irlmproc,SET 775
Format 775
Usage. 776
Examples 778
F irlmproc,STATUS 779
Format 779
Usage. 779
Examples 780

Chapter 73. STOP CQSJOB 783
Format 783
Usage. 783

Chapter 74. STOP IRLMPROC 785
Format 785
Usage. 785
Example 785

Chapter 75. TRACE CT 787
Format 787
Usage. 787
Ensure IRLM Load Module Is in z/OS Link List. 789
Sample External Writer Procedure 789
Displaying a Trace 789
Examples 789
Example 1 for TRACE CT 789
Example 2 for TRACE CT 789

Chapter 76. CANCEL/FORCE ODBA 791
Format 791
Usage. 791
Format 791
Usage. 791

Chapter 77. STOP CSL Address Spaces 793
Format 793
Usage. 793

Chapter 67. Introduction

This section covers z/OS commands that are used by the z/OS console operator to control the execution of the following:

- CQS
- CSL
- FDBR
- IRLM
- ODBA

© Copyright IBM Corp. 1974, 2003

Chapter 68. START FDBRPROC

Format

Usage

Although you normally start an IMS Fast DB Recovery region using a job submitted after the databases to be tracked are started, you can use the z/OS START command to start the region and tracking.

The fdbrproc parameter identifies the procedure name of the region to be started.

© Copyright IBM Corp. 1974, 2003

Chapter 69. MODIFY IMS

You can use the following z/OS commands to control an IMS region.

F jobname,DUMP

Format

Usage

The MODIFY (F) jobname,DUMP command forces a termination of IMS with an offline dump.

For DBCTL, the following are abended, but are not rolled back until the following /ERESTART command:

- Active threads
- BMPs
- DEDB utilities

Example

Entry z/OS SC:

```
F jobname,DUMP
```

Response z/OS SC:

```
DFS628I ABNORMAL TERMINATION SCHEDULED
DFS629I IMS STM TCB ABEND - IMS 0020
```

F jobname,DUMPxxxx

Format

Usage

Note: “External subsystem” means a non-CCTL subsystem.

The MODIFY (F) jobname,DUMP*xxxx* command causes one or more attached external subsystem connections to be terminated with a dump. If *xxxx* is specified as ESS, the parent external subsystem task and all external subsystem subtasks are terminated. If *xxxx* is a specific external subsystem name, only that external subsystem is terminated. The subsystem specified must be defined in the IMS external subsystem PROCLIB member.

© Copyright IBM Corp. 1974, 2003

Examples

Example 1 for F jobname,DUMPxxxx

Entry z/OS SC:

```
F jobname,DUMPname
```

Response z/OS SC:

```
DFS628I ABNORMAL TERMINATION SCHEDULED
DFS629I IMS ESI TCB DUMP - IMS 0020 IES
DFS3611I EXTERNAL SUBSYSTEM name CONNECTION TERMINATED
```

Example 2 for F jobname,STOPxxxx

Entry z/OS SC:

```
F jobname,DUMPESS
```

Response z/OS SC:

```
DFS628I ABNORMAL TERMINATION SCHEDULED
DFS629I IMS ESS TCB DUMP - IMS 0020
DFS629I IMS ESS TCB DUMP - IMS 4095 IES
DFS3611I EXTERNAL SUBSYSTEM name CONNECTION TERMINATED
```

F jobname,FORCExxxx

Format

Usage

The MODIFY (F) jobname,FORCE*xxxx* command causes one or more attached external subsystem connections to be terminated without a dump. If *xxxx* is specified as ESS, the parent external subsystem task and all external subsystem subtasks are terminated. If *xxxx* is a specific external subsystem name, only that external subsystem is terminated. The subsystem specified must be defined in the IMS external subsystem PROCLIB member.

The FORCE option is not applicable to any other IMS task, for example, to the control task. IMS does not communicate with the external subsystem termination exits, for example, terminate subsystem exits and terminate identify exits.

The reason IMS does not communicate is to allow the installation to break the IMS-to-other-subsystem connection in the event of an error (such as a WAIT STATE) in the external subsystem. If IMS were to invoke one of the exits, as it does with the STOP/DUMP options, the exit can once again start a WAIT STATE, and not allow the termination of that subsystem connection.

Use the FORCE option when either STOP or DUMP fails to break the subsystem connection.

Examples

Example 1 for F jobname,FORCExxxx

Entry z/OS SC:

```
F jobname,FORCEname
```

Response z/OS SC:

```
DFS628I ABNORMAL TERMINATION SCHEDULED
DFS629I IMS ESI TCB DUMP - IMS 0020 IES
DFS3611I EXTERNAL SUBSYSTEM name CONNECTION TERMINATED
```

Example 2 for F jobname,FORCExxxx

Entry z/OS SC:

```
F jobname,FORCEESS
```

Response z/OS SC:

```
DFS628I ABNORMAL TERMINATION SCHEDULED
DFS629I IMS ESS TCB DUMP - IMS 0020
DFS629I IMS ESS TCB DUMP - IMS 4095 IES
DFS3611I EXTERNAL SUBSYSTEM name CONNECTION TERMINATED
```

F jobname,RECONNECT

Format

Usage

The MODIFY (F) jobname,RECONNECT command causes the IMS system on which it is entered to be reconnected to a restarted IRLM.

Jobname is the job name or procedure name for the IMS control region. When this command is successfully completed, a message indicating that the IRLM has been reconnected is sent to both the IMS master terminal operator and the z/OS console operator. Then the IMS MTO can start any Fast Path DEDB areas that were involved in block-level sharing at the time of the IRLM failure.

Example

In this example, two IMS subsystems are connected to the same IRLM when it terminates.

z/OS SYSTEM SC RECEIVES:

```
DXR121I KRLM212 END-OF-TASK CLEAN-UP SUCCESSFUL - HI-CSA  410K -
   HI-ACCT-CSA   0K
```

IMS 1 MT RECEIVES:

```
DFS2500I IMSA DATABASE dbname SUCCESSFULLY DEALLOCATED
```

IMS 2 MT RECEIVES:

```
DFS2500I IMSB DATABASE dbname SUCCESSFULLY DEALLOCATED
```

z/OS SYSTEM SC:

```
S KRLM1
```

z/OS SYSTEM SC RECEIVES:

```
DXR180I KRLM212 AUTOMATIC RESTART MANAGER IS NOT ENABLED
                OR
     DXR172I KRLM212 armelementname  ARM READY COMPLETED

     DXR117I KRLM212 INITIALIZATION COMPLETE
```

z/OS SYSTEM SC:

```
F jobname,RECONNECT
```

IMS 1 MT AND z/OS SYSTEM SC RECEIVE:

```
DFS626I KRLM RECONNECT COMMAND SUCCESSFUL. IMSA
```

z/OS SYSTEM SC:

```
F jobname,RECONNECT
```

IMS 1 MT AND z/OS SYSTEM SC RECEIVE:

```
DFS626I KRLM RECONNECT COMMAND SUCCESSFUL. IMSB
```

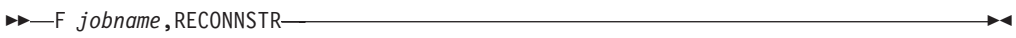
IMS MT:

```
/START AREA areaname
```

Explanation: The IRLM has abnormally terminated. When the IRLM has been successfully restarted, the z/OS operator modifies the IMSs to reconnect to the restarted IRLM. The areas can then be restarted using the /START AREA command.

F jobname,RECONNSTR

Format



Usage

Use this command to cause the IMS subsystem on which you enter it to reconnect to IRLM, OSAM, and VSAM coupling facility structures.

jobname is the job name or procedure name for the IMS control region.

Example

Entry z/OS SC:

```
F jobname,RECONNSTR
```

Explanation: If the connection to IRLM, OSAM, or VSAM structures has abnormally terminated, usually IMS automatically reconnects to these structures. If IMS does not reconnect to the structures, the operator can use this command to ensure IMS reconnects to them.

F jobname,STOP

Format

Usage

Active threads, BMPs, and DEDB utilities are abended, but not rolled back until the following /ERESTART command.

Normally, IMS is terminated with an IMS /CHECKPOINT shutdown command. Whenever it becomes necessary to force termination of IMS, the z/OS MODIFY command should be used. When MODIFY is used, a dump of the IMS control region can be requested. (Consult the appropriate z/OS operator's guide for specific MODIFY formats.)

The MODIFY (F) jobname,STOP command forces a termination of IMS without a dump.

Example

Entry z/OS SC:

```
F jobname,STOP
```

Response z/OS SC:

```
DFS628I ABNORMAL TERMINATION SCHEDULED
DFS629I IMS STM TCB ABEND - IMS 0020
```

F jobname,STOPxxxx

Format

Usage

Note: All references to external subsystems refer to non-CCTL subsystems.

The MODIFY (F) jobname,STOP*xxxx* command causes one or more attached external subsystem connections to be terminated without a dump. If *xxxx* is specified as ESS, the parent external subsystem task and all external subsystem subtasks are terminated. If *xxxx* is a specific external subsystem name, only that external subsystem is terminated. The subsystem specified must be defined in the IMS external subsystem PROCLIB member.

Examples

Example 1 for F jobname,STOPxxxx

Entry z/OS SC:

```
F jobname,STOPname
```

Response z/OS SC:

```
DFS628I ABNORMAL TERMINATION SCHEDULED
DFS629I IMS ESI TCB DUMP - IMS 0020 IES
DFS3611I EXTERNAL SUBSYSTEM name CONNECTION TERMINATED
```

Example 2 for F jobname,STOPxxxx

Entry z/OS SC:

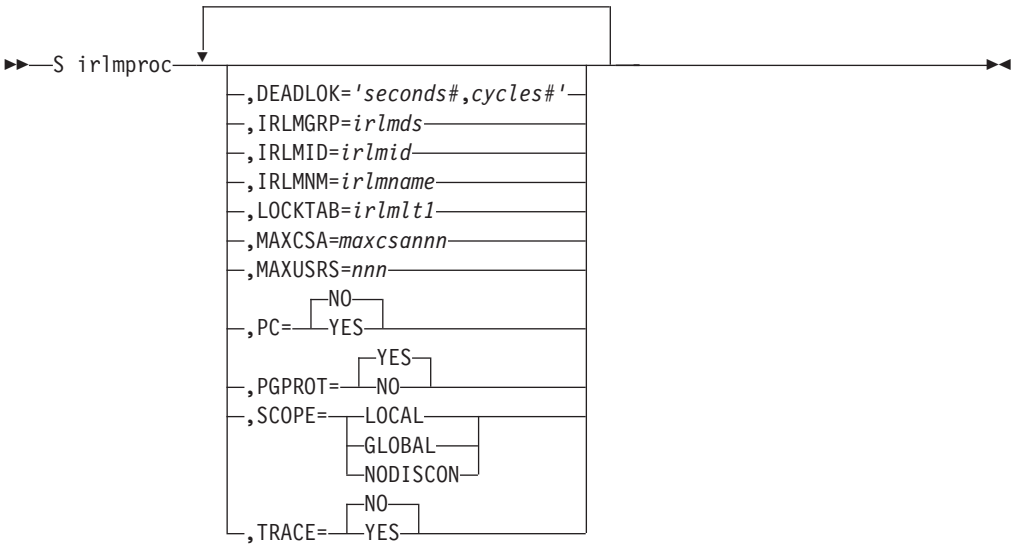
```
F jobname,STOPESS
```

Response z/OS SC:

```
DFS628I ABNORMAL TERMINATION SCHEDULED
DFS629I IMS ESS TCB DUMP - IMS 0020
DFS629I IMS ESS TCB DUMP - IMS 4095 IES
DFS3611I EXTERNAL SUBSYSTEM name CONNECTION TERMINATED
```

Chapter 70. START IRLMPROC

Format



Usage

The START irlmproc command starts an IRLM component with a procedure put in place by your installation. Symbolic parameters in the procedure can be overridden using the START irlmproc command. IRLM must be started before any IMS subsystem that needs the facilities provided by IRLM.

The irlmproc identifies the procedure name of the IRLM to be started.

Any IRLM parameter can be overridden using the START command.

DEADLOK=
Specifies the local deadlock-detection interval (in seconds), and the number of local cycles that are to occur before a global detection is initiated. You can specify the following:

seconds#
This 1- to 4-digit numeric value, ranging from 1 to 9999, specifies the interval (in seconds) between IRLM local deadlock-detection cycles. Although values above 5 are accepted, the maximum used is 5.

Recommendation: Set this parameter to be no greater than 5. This is because it takes multiple local cycles to complete one global cycle. Otherwise, global deadlocks may not be broken frequently enough to ensure maximum throughput.

cycles#
This 1- to 4-digit numeric value, ranging from 1 to 9999, specifies the number of local deadlock cycles that must expire before global deadlock-detection is performed. Although values above 1 are accepted, the maximum used is 1.

© Copyright IBM Corp. 1974, 2003

Recommendation: Set this parameter to be no greater than 5. This is because it takes multiple local cycles to complete one global cycle. Otherwise, global deadlocks may not be broken frequently enough to ensure maximum throughput.

IRLMGRP=
Specifies the name of the z/OS Cross-System Coupling Facility (XCF) group for which this IRLM will serve as lock manager.

irlmds
The name of the XCF group.

IRLMID=
Specifies either a decimal number or a printable character that is used to distinguish between multiple IRLMs. The IRLM with the lowest ID value in the group becomes the global deadlock manager for the group. You must specify a unique name for every IRLM in the group. The total number of IRLMs that can be included in a group is not related to the value you specify; the total number is limited by your hardware, and may be lower than 255.

The *irlmid* can be either a one- to three-digit number ranging in value from 1 to 255, or a printable EBCDIC character. When you specify a character, you must surround the character with enough single quotes to permit IRLM to understand that this value is indeed a printable character. IRLM converts the character to an EBCDIC value between 1 and 255.

Because of the way that z/OS interprets quotes, you must use 7 quotes on either side of the character.

Example: If you want to specify the printable character 'D', you must specify `IRLMID='''''''D'''''''`.

IRLMNM=
Specifies the 1- to 4-byte z/OS subsystem name assigned to this IRLM.

LOCKTAB
Specifies the lock table to be used by the data sharing group specified in the GROUP parameter.

irlmlt1
Is the name of the lock table.

MAXCSA=
Specifies the maximum amount of CSA that IRLM is to use for its lock structures. In displays, this storage is called *accountable* storage because it is accountable to the value you set for MAXCSA. Use the accountable storage high water mark (AHWM) data from messages DXR100I and DXR121I to monitor IRLM's usage of common storage and as a basis for adjustments to the MAXCSA value.

This parameter must be specified as a 1 or 3-digit number from 1 to 999. This number indicates the number of megabytes of CSA storage that IRLM will use. If IRLM is using extended CSA support and running on a z/OS system, this number indicates multiples of 1 MB of extended CSA storage. For example, a specification of 3 states that IRLM can use 3 MB of CSA, or 3 MB of ECSA, if IRLM is running with extended CSA support on a z/OS system.

MAXUSRS=
Specifies the maximum number of users, from 2 to 248, that can connect to the data sharing group.

PC=

Specifies whether a space switching program call is to be used by IRLM. If PC=YES is specified, the IRLM lock control block structure resides in the IRLM address space, and the space switching program call (PC) instruction is used to obtain addressability to the structure. If PC=NO is specified, the IRLM lock structure resides in ECSA. The default for this parameter is PC=NO. Tells IRLM whether to place its COMMON STORAGE load modules into z/OS Page Protected Storage. PGPROT=YES is the DEFAULT and tells IRLM to place its COMMON Storage load modules into z/OS Page Protected Storage. If PGRPROT=NO is specified, IRLM will not place load modules in Page Protected Storage.

SCOPE=

Specifies whether the IRLM is to be used in a data-sharing environment and, if so, whether the IRLM should automatically disconnect when the associated IMS terminates.

LOCAL

Specifies that the IRLM is in a non-data-sharing environment and there is no intersystem sharing.

GLOBAL

Specifies that the IRLM is in a data-sharing environment and that intersystem sharing is to be performed. The IRLM disconnects from the data-sharing group when there are no IMSs identified to it. Both XCF and SLM are required

NODISCON

Specifies that the IRLM is in a data-sharing environment and that intersystem sharing is to be performed. The IRLM remains connected to the data-sharing group even when there are no IMSs identified to it. Normal DISCONNECT from the sharing group does not occur. You must explicitly stop the IRLM when `NODISCON` is specified. Both XCF and SLM are required.

With `NODISCON`, there is less impact on other systems when an IMS fails because z/OS is not required to perform certain recovery actions that it normally perform when an IRLM terminates. Another benefit of the `NODISCON` parameter is that IMS restarts more quickly after a normal or abnormal termination because it does not have to wait for IRLM to rejoin the IRLM data-sharing group.

TRACE=

Specifies whether the IRLM should trace activity during IRLM startup. Traces are written to wrap-around trace buffers, that is, the first buffer is reused after the last buffer is full.

NO

Do not initialize trace activity during IRLM startup. `NO` is the default.

To trace IRLM activity, use the `TRACE CT` command. See Chapter 75, “TRACE CT,” on page 787

YES

Initialize IRLM trace activity during startup.

Recommendation: Turn on tracing in all data-sharing environments.

Chapter 71. MODIFY FDBRPROC

The `MODIFY fdbrproc` commands are used for an IMS Fast Database Recovery region: to display its status, to stop it, and to recover from failures.

F fdbrproc,DUMP

Format

Usage

Use this command to stop IMS Fast DB Recovery functions and request a dump for the region. The region ends with a return code X'0020'.

F fdbrproc,RECOVER

Format

Usage

Use this command to initiate an IMS Fast DB Recovery of tracked IMS databases. Use this command, for example, after a time-out during XCF or log surveillance.

This command is not accepted if an IMS Fast DB Recovery region is currently starting or if it is recovering databases.

F fdbrproc,STATUS

Format

Usage

Use this command to display status for an IMS Fast DB Recovery region. The output displayed includes the following information:

PHASE This field indicates the phase of the Fast DB Recovery region. The following are possible values for the phase:

- INIT
- TRACKING
- RECOVERY

LOG-TIME Time of the log record currently being processed by the Fast DB Recovery region. The length of time

© Copyright IBM Corp. 1974, 2003

the Fast DB Recovery region lags behind IMS in reading the log is the difference between the current time, as shown by the time stamp, and the log time. This field displays N/A during FDBR initialization.

ACT-ID The IMSID of the IMS subsystem that Fast DB Recovery is tracking. This field displays N/A during FDBR initialization.

GROUPNAME XCF group name used for XCF monitoring. This field displays N/A during FDBR initialization.

TIMEOUT XCF timeout value. This field displays N/A during FDBR initialization.

AREA01 The sharing level for DEDB areas. The following are possible values for AREA01:

- RECOV
- NORECOV

You specify the sharing level for AREA01 in the DFSFDRxx member of IMS.PROCLIB.

3 **SVSOOPEN** Specifies that all areas requiring redo processing in an FDBR system will be serially processed. This option is ignored for /ERE and XRF TKO processing. It is used to reduce the number of structures being allocated by FDBR for redo processing.

3 **FPBUFF** Specifies that the control blocks for Fastpath DEDB processing are to be gotten from the FDBR private region rather than ECSA.

The LOCAL parameter is specified in the DFSFDRxx member. If this option is selected, the control blocks for DEDB processing are taken from the FDBR control region private storage.

If this option is not specified, the control blocks for DEDB processing are taken from ECSA. This is the default.

3 **Example:**

```
3 F FDR1,STATUS
3 DFS000I PHASE: TRACKING  LOG-TIME: 17:48:41  FDR1
3 DFS000I     ACT-ID: SYSS3    GROUPNAME: FDRSYS3  FDR1
3 DFS000I TIMEOUT: 060 SEC AREA01: NORECOV     FDR1
3 DFS000I SVSOOPEN: SERIAL FPBUFF: LOCAL       FDR1
```

F fdbrproc,STOP

Format

Usage

Use this command to stop an IMS Fast DB Recovery region. The region ends with a return code X'0020'.

F fdbrproc,TERM

Format

Usage

Use this command to stop IMS Fast DB Recovery tracking activity.

This command is not accepted if an IMS Fast DB Recovery region is currently starting or if it is recovering databases.

Chapter 72. MODIFY IRLMPROC

The MODIFY (F) irlmproc commands are used to display the status of an IRLM and abnormally terminate an IRLM.

These commands can be issued only from a z/OS console.

F irlmproc,ABEND

Format

Usage

The MODIFY irlmproc,ABEND command is used to abnormally terminate IRLM whether or not any IMS subsystems are identified to IRLM.

The *irlmproc* identifies the procedure name of the IRLM to be terminated. Prior to terminating, IRLM initiates all active IMS STATUS exit routines to inform them of the pending termination.

If the NODUMP parameter is specified, IRLM terminates without generating a dump. If the NODUMP parameter is not specified, the system dump is taken to the SYS1.DUMPxx data set.

Recommendation: If IRLM does not terminate because of outstanding IMS requests in process (the DXR011I irlmproc END-OF-TASK CLEAN-UP SUCCESSFUL message is not received), use the following methods (in this order) to terminate IRLM:

1. Issue the MODIFY irlmproc,ABEND,NODUMP command to terminate IRLM without attempting to clean up the outstanding IMS requests.
2. Issue the z/OS CANCEL command.
3. Issue the z/OS FORCE command. This method is least preferred because it might prevent the irlmproc from completing the cleanup required to permit the subsystem to be restarted.

Examples

Example 1 for F irlmproc,ABEND

z/OS SYSTEM 1 SC:

```
F IR21J,ABEND
```

Response SYSTEM 1 SC: (partial messages)

```
DXR124E JR21212 ABENDED VIA MODIFY COMMAND
DUMPID=001 REQUESTED BY JOB (IR21J   )
IEA794I SVC DUMP HAS CAPTURED: 857
DUMP TITLE=JR21 ESTAE ENTERED. ABEND U2020 MODULE DXRRL020+0A98
     APAR PQ15432  1999/007 14:14:03
IXL030I CONNECTOR STATISTICS FOR LOCK STRUCTURE LOCK2, 860
  CONNECTOR IRLMGRPR$JR21212:
```

© Copyright IBM Corp. 1974, 2003

```
IXL031I CONNECTOR CLEANUP FOR LOCK STRUCTURE LOCK2, 861
  CONNECTOR IRLMGRPR$JR21212, HAS COMPLETED.
DXR121I JR21212 END-OF-TASK CLEANUP SUCCESSFUL - HI-CSA    410K  -
HI-ACCT-CSA      0K
IEF450I IR21J IR21J - ABEND=S000 U2020 REASON=00000000
IEF352I ADDRESS SPACE UNAVAILABLE
$HASP395 IR21J    ENDED
```

Response SYSTEM 2 SC: (partial messages)

```
DXR137I IR21001 GROUP STATUS CHANGED. JR21 212 HAS BEEN DISCONNECTED
FROM THE DATA SHARING GROUP
```

Explanation: The operator on system 1 has terminated the procedure named IR21J. The operator on system 2 is informed that the IRLM on system 1 has failed.

Example 2 for F irlmproc,ABEND

z/OS SYSTEM 2 SC:

```
F IR21J,ABEND,NODUMP
```

Response SYSTEM 2 SC: (partial messages)

```
DXR165I JR21212 TERMINATED VIA IRLM MODIFY COMMAND.
  IXL030I CONNECTOR STATISTICS FOR LOCK STRUCTURE LOCK2, 976
  CONNECTOR IRLMGRPR$JR21212:
  ....
   IXL020I CLEANUP FOR LOCK STRUCTURE LOCK2, 980
      CONNECTION ID 02, STARTED BY CONNECTOR IRLMGRPR$IR21001
   IXL021I GLOBAL CLEANUP FOR LOCK STRUCTURE LOCK2, 981
      CONNECTION ID 02, BY CONNECTOR IRLMGRPR$IR21001
      HAS COMPLETED.
   DXR121I JR21212 END-OF-TASK CLEANUP SUCCESSFUL - HI-CSA    343K  -
      HI-ACCT-CSA      0K
   IEF352I ADDRESS SPACE UNAVAILABLE
   $HASP395 IR21J    ENDED
```

Response SYSTEM 1 SC: (partial messages)

```
DXR137I IR21001 GROUP STATUS CHANGED. JR21 212 HAS BEEN DISCONNECTED
      FROM THE DATA SHARING GROUP
```

Explanation: The operator on system 2 has terminated the procedure named IR21J and has suppressed the SDUMP dump. The operator on system 1 is informed that the IRLM on system 2 has failed.

F irlmproc,DIAG,HANG

Format

Usage

HANG is used under the direction of IBM service for collecting IRLM SYSPLEX dumps when DEADLOCK or TIMEOUT issues are suspected.

This command takes dumps during DEADLOCK processing and causes DEADLOCK processing to STOP while the dynamic deadlock storage is collected. z/OS DUMP services then schedules an SRB to restart DEADLOCK processing. DXR183I is issued by each IRLM as the SRB resumes IRLM deadlock.

If this message is not issued, that IRLM must be terminated and restarted.

Note: Always start the IRLM SCF CTRACE internally and wait 30 seconds before issuing this command.

F irlmproc,PURGE,imsname

Format

Usage

The MODIFY (F) irlmproc, PURGE,imsname command releases IRLM locks retained for a specific IMS subsystem because of an IMS, IRLM, or system failure. The command causes STATUS exit routines to be activated for each identified IMS. This command is used in the following recovery situations:

- Database cleanup was performed and reflected in the RECON data set, but the IRLM was not made aware of it. This typically occurs when a batch backout is performed before the IRLMs are reconnected. After the PURGE command is issued, data that was locked is made available to the active IMSs.
- Recovery will be performed later and the databases must be unavailable for use by other IMSs.

Restriction: Ensure the IMS Database Manager is inactive prior to issuing this command. In a DBCTL environment, use the /CHANGE CCTL command when the DBMS is active.

The *imsname* is the IMS name displayed by the STATUS command.

The *irlmproc* identifies the procedure name for each IRLM. If multiple IRLMs exist in the same system, each procedure must have a unique procedure name.

Example

Entry z/OS SC:

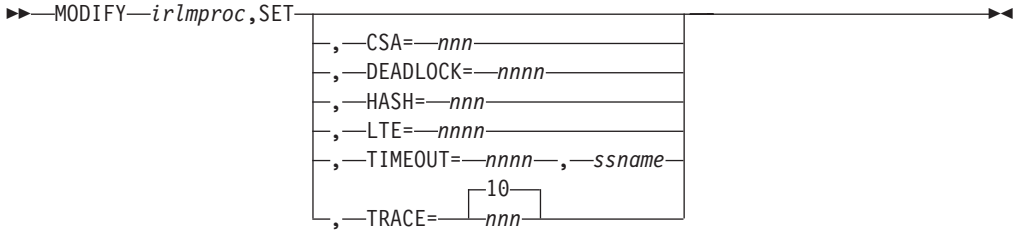
```
F IR21I,PURGE,IMS1
```

Response SYSTEM 1 SC:

```
DXR109I IR21001 PURGE COMMAND COMPLETED FOR IMS1
```

F irlmproc,SET

Format



Usage

The `Modify (F) irlmproc,SET` command dynamically alters a limited set of parameters known to IRLM.

irlmproc
Specifies the IRLM that is to process the command.

CSA=*nnn*
Requests that IRLM dynamically sets the maximum amount of CSA that this IRLM can use for lock control structures. These structures are allocated from ECSA when `PC=NO`. This command is ignored if `PC=YES`. Valid values are from 1 through 999 (megabytes). IRLM does not immediately allocate ECSA storage for the new value you set in this command. IRLM allocates storage as needed, not to exceed the amount of ECSA specified in the command. If the amount of storage currently allocated by IRLM is greater than the amount of ECSA you specify in this command, more storage is not obtained until normal processing frees enough storage to bring the current allocation below the new ECSA value you set.

DEADLOCK= *nnnn*

nnnn The number in milliseconds to be used for the local deadlock frequency. The value range is 100 - 5000 milliseconds. If a member of a sysplex group and all IRLMs are not enabled for subsecond deadlock processing, DXR106E will be issued.

When the IRLM supporting subsecond deadlock joins a group which has a member which does not support subsecond deadlock, the value range used for the new member joining, is 1 to 5 seconds. All members supporting subsecond processing will issue the DXR177I message after changing the timer value.

Note: Once IMS TIMEOUT candidates have timed out, they remain timeout candidates and are presented to the timeout exit each Global deadlock cycle. IMS creates SMF 79.15 records when candidates are presented. They are then written to the SMF datasets, if enabled. If timeout candidates are found and the value for *nnnn* is subsecond, there will be many SMF 79.15 records written per second until the tasks are no longer waiting in IRLM.

HASH=*nnnn*
Determines the number of LOCK HASH entries to be specified on the next connect to the XCF LOCK structure. Valid values must be an even power of two between 0-1024. Each increment in value represents 1,048,576 hash entries.

Any syntax error in issuing the command will receive DXR106E. Syntax errors include HASH value out-of-range or invalid identified subsystem name. If IRLM is not connected to the group and the value is valid, the DXR177I message will be issued but the value will not be sent to any other member. If the member is already in the group, the value is sent to the Global Deadlock Manager IRLM to be broadcast to all other members. If the GDM does not have the code applied, no DXR177I response will be issued on any member. If the GDM has the code, then all members with the code applied will issue the DXR177I as the command is processed. This value is only used if the IRLM is the first to join the data sharing group causing structure allocation, or during a REBUILD. Any IRLM joining later will not have the updated value. If multiple `MODIFY` commands are issued, some DXR177I messages may be missing. The last DXR177I issued is the value to be used on the next CONNECT.

The value for the number of LOCK HASH entries specified during normal group startup and during REBUILD are used in the following order:

1. The value specified on the `MODIFY irlmproc,SET,HASH=` command if it is greater than 0.
2. The value from the HASH= in the IRLMPROC if it is greater than 0
3. The existing logic, which determines the nearest power of 2 after dividing the QUERY size returned by 2 times Hash width based on MAXUSRS.

If an attempt is made to use a nonzero value from either 1 or 2 above and that value is too large for the structure size returned on the QUERY, then the value from the next lower order sequence is used.

LTE=*nnnn*

nnnn The number of Lock Table Entries to be specified on the next CONNECT to the XCF LOCK structure. The value range must be 0 - 1024 and be an even power of two. Each increment in value represents 1,048,576 Lock Table Entries.

Any syntax error in issuing the command will receive DXR106E. Syntax errors include LTE value out-of-range. if this IRLM is not connected to the group, and the value is valid, it will issue DXR177I, but the value will not be sent to any other member. If the member is already in the group, the value is sent to the Global Deadlock Manager (GDM) IRLM to be broadcast to all other members. If the GDM does not have the code applied, no DXR177I response will be issued on any member. If the GDM has the code, then all members with the code applied will issue the DXR177I as the command is processed.

This value is only used if the IRLM is the first to join the data-sharing group causing structure allocation or during a REBUILD. If any IRLM joins later, they will not have the updated value. If multiple `MODIFY` commands are issued on the same or multiple IRLMs, some DXR177I may be missing. The last DXR177I issued is the value to be used on the next CONNECT.

The value for the number of Lock Table Entries specified during normal group startup and during REBUILD are used in the following order:

1. The value specified on the `MODIFY irlmproc,SET,LTE=` command if it is greater than zero.
2. The value from the LTE= in the IRLMPROC if it is greater than zero.
3. The existing logic, which determines the nearest power of 2 after dividing the QUERY size returned by 2 times LTE width based on MAXUSRS.

If an attempt is made to use a nonzero value from either step 1 or 2 and that value is too large for the structure size returned on the QUERY, then the value from the next lower order sequence is used.

TIMEOUT=*nnnn,ssname*

nnnn The new timeout value desired between 1 and 3600 seconds.

ssname
The identified subsystem name as displayed by the `MODIFY irlmproc,STATUS` command.

Any syntax error in issuing the command will receive DXR106E. Syntax errors include TIMEOUT value out-of-range or invalid identified subsystem name. A

syntax error message will also be given if the DXR177I message has not been received for the prior command completion.

The TIMEOUT value must be a multiple of the local deadlock parameter. If the value entered is not an even multiple of the local deadlock parameter, IRLM increases the timeout value to the next highest multiple. The value used by IRLM for timeout is displayed in the DXR177I message, which is issued during deadlock processing. This new value is used until the IRLM or identified subsystem is terminated, or the timeout is changed again by the operator. The value specified on the command does not affect the time out value in the DB2 ZParms.

TRACE=*nnn*
Requests that IRLM dynamically sets the maximum number of 64KB trace buffers per trace type to the value you specify in *nnn*. This value is used only when the external CTRACE writer is not activated. The trace buffers are allocated from ECSA. Valid values are from 10 through 255. If you specify a value outside of this range, IRLM automatically adjusts the value to a value within the range. The default is 10. IRLM does not immediately acquire the number of trace buffers you set, but allocates buffers as needed, not to exceed the number of buffers you set in this command. If the number of trace buffers you set is less than the number of currently allocated buffers, IRLM brings the number to within your specified range by releasing the oldest buffers at the end of the next deadlock or timeout cycle.

Restrictions:

1. Do not modify the CSA value without first contacting the system programmer to determine the amount of CSA storage that can be used for IRLM.
2. The values you set in the MODIFY command do not persist through a stop and restart of IRLM. The number of trace buffers for each trace type returns to the default value of 10 and the value for MAXCSA returns to the value you set for the MAXCSA parameter of the IRLM startup procedure.

Examples

Example 1 for F irlmproc,SET

Entry on a z/OS1 system console:

```
F IRLMPROC,SET,CSA=10
```

Response on z/OS1 system console:

```
DXR177I IRLMX THE VALUE FOR MAXCSA IS SET TO 10
```

Example 2 for F irlmproc,SET

Entry on a z/OS system console:

```
F IRLMPROC,SET,TRACE=20
```

Response on system console:

```
DXR177I IRLMX THE VALUE FOR TRACE IS SET TO 20
```

Example 3 for F irlmproc,SET

Entry on a z/OS system console:

```
F IRLMPROC,SET,TIMEOUT=60,DBMS
```

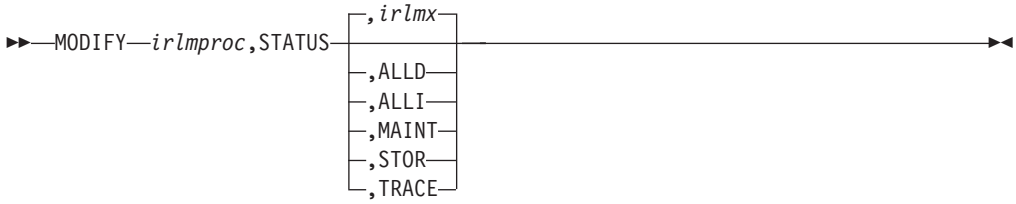
Response on system console:

```
DXR177I IRLMX THE VALUE FOR TIMEOUT IS SET TO 60 FOR DBMS
```

F irlmproc,STATUS

This command is used to display the status of an IRLM.

Format



Usage

The `MODIFY (F) irlmproc,STATUS` command is used to display the status of an IRLM.

The *irlmproc* identifies the IRLM that is to process the command.

irlmx
Specifies which IRLM's status is to be displayed. The *irlmx* parameter is the concatenation of the IRLM subsystem name (IRLMN) and a three-digit IRLM member ID (IRLMID) specified in the IRLM startup procedure. An example is DB2G002 (ID is 002).

ALLD
Requests the DBMS names and status of subsystems, active or failed, on any IRLM in the data sharing group. This keyword also shows any RETAINED LOCKS the systems might own and the IRLM they are currently associated with.

ALLI
Requests the IRLM names and status for all IRLMs in a data sharing group.

MAINT
Allows the operator to display the IRLM maintenance levels for IRLM load modules, with the exceptions of: DXRRLM50, DXRRL183, DXRRL186, DXRRLFTB, and in a non-sysplex environment DXRRLM70.

STOR
For this IRLM, displays the current and maximum specified allocation for CSA and ECSA storage.

TRACE
For this IRLM, requests information about IRLM subcomponent trace types. The information returned includes whether a subcomponent trace type is active, how many trace buffers are used by the trace, and whether the component trace external writer is active for the trace.

You can activate or deactivate traces by using the `TRACE CT` command of z/OS. You cannot turn off the EXP and INT traces. The XIT (for data sharing), EXP, and INT traces are automatically activated when you start IRLM. Traces are automatically activated with `IRLMPROC TRACE=YES`.

Message DXR001I is issued if the *irlmx* parameter is omitted or specifies the IRLM receiving the command. This multiline message contains work unit and lock information for each IMS identified to this IRLM.

Message DXR001I is also issued if the *irlmx* parameter is specified, but the IRLM receiving the command has no knowledge of an IRLM with an IRLMID of *irlmx*.

Message DXR102I is issued if *irlmx* is specified. This multiline message contains the IMS names and status of each IMS subsystem identified to the IRLM with an IRLMN and IRLMID of *irlmx*. Message DXR102I is also issued if ALLD is specified. It contains the names and status of each IMS subsystem identified to the IRLM in the data sharing group.

Message DXR103I is issued if ALLI is specified. This multiline message contains the IRLM names and status of all IRLMs in the data sharing group.

Examples

Example 1 for F irlmproc,STATUS

Assume you are in a non-data sharing environment. Enter on the z/OS1 system console:

```
F IRTPROC,STATUS
```

Response on z/OS1 system console:

```
DXR101I IR2T001 STATUS SCOPE=LOCAL
        SUBSYSTEMS IDENTIFIED               PT01
        NAME      STATUS    UNITS     HELD      WAITING     RET_LKS
        DSNT1     UP-NS     0005      0010      0002        0
```

Explanation: The operator on system 1 has requested information about the IMS systems connected to the IRLM identified by the IRLM procedure named IRTPROC.

Example 2 for F irlmproc,STATUS

Assume you have a data sharing group. Enter on a system console:

```
F DB1GIRLM,STATUS,ALLD
```

Response on system console:

```
  14.02.10 STC00086  DXR102I DJ1G001 STATUS IRLMID=001
          SUBSYSTEMS IDENTIFIED                PT01
          NAME     STATUS    RET_LKS   IRLMID  IRLM_NAME
          DB4G     UP              0    004      DJ4G
          DB3G     UP              0    003      DJ3G
          DB2G     UP              0    002      DJ2G
          DB1G     UP              0    001      DJ1G
```

Explanation: The output shows all the IMS subsystems that are connected to IRLMs in this data sharing group (the group to which the IRLM processing the request belongs). The value “UP” in the STATUS field indicates that the IMS is active. Other possible values for STATUS include:

DOWN The IMS is failed.

UP-RO IMS is active and is currently identified to IRLM as a read-only subsystem.

CLEANUP IRLM is waiting for the IMS subsystem to respond that cleanup has completed after a failure condition.

SFAIL The IRLM that IMS is identified to has been disconnected from the data sharing group. Any “modify” type locks held by IMS have been retained by IRLM.

Example 3 for F irlmproc,STATUS

Again, assume data sharing is in effect. Enter the following on the system console:

```
F DB1GIRLM,STATUS,ALLI
```

The response on the console is:

```
  14.03.50 STC00086  DXR103I DJ1G001 STATUS IRLMID=001
          IRLMS PARTICIPATING IN DATA SHARING  PT01
          IRLM_NAME   IRLMID   STATUS
            DJ4G       004       UP
            DJ3G       003       UP
            DJ2G       002       UP
            DJ1G       001       UP
```

Explanation: The output shows the IRLMs that are participating in this data sharing group (the group that includes the IRLM processing the request). The value “UP” in the STATUS field indicates that the IRLM is active. STATUS shows “DOWN” if the IRLM is failed.

Example 4 for F irlmproc,STATUS

Enter the following command on the system console:

```
F IR21PROC,STATUS,STOR
```

The response on the console is:

```
DXR1001 IR21001 STOR STATS
   PC:NO  MAXCSA:   6M[1]
CSA USE:ACNT: 132K  AHWM: 132K[2] CUR: 4048K HWM: 4086K[3]
   ABOVE 16M:  72 4033K   BELOW 16M:   6   15K
CLASS  TYPE SEGS  MEM  TYPE SEGS  MEM  TYPE SEGS  MEM
ACCNT  T-1    1   64K  T-2    1   64K  T-3    1    4K[4]
PROC   WRK   11   58K  SRB    3    3K  OTH    2    2K
MISC   VAR   60 4081K  N-V    6   22K  FIX    1   24K[5]
```

Notes:

1. Displays the current values for the PC and MAXCSA options of the IRLM startup procedure.
2. Shows storage use of accountable storage; that is, storage that is accountable toward the MAXCSA value of the IRLM procedure. In this output, the current use accountable storage (ACNT) is 132 KB. The high water mark since the last time IRLM was started is also 132 KB.
3. Shows the total current CSA and ECSA usage. In this case, the current usage is 4048 KB and the high water mark is 4086 KB. The accountable storage is a subset of this total storage.
4. The ACCNT row of the report is a breakdown of lock control block structures and their storage use.

   T-1 Type 1 structures are for resources. In this case, it shows that one storage segment is held for a total of 64 KB.

   T-2 Type 2 structures are for all resource requests after the first request for a specific resource. In this case, it shows that one storage segment is held for a total of 64 KB.

   T-3 Type 3 structures are for requesters (or work units) that are waiting for resources or are currently holding resources. In this case, it shows that one storage segment is held for a total of 4 KB.
5. The PROC and MISC rows contain usage information for CSA, ECSA, and private storage used to process DBMS requests. Use this information under the guidance of IBM service for diagnosing problems.

Example 5 for F irlmproc,STATUS

In this example, the response on the console displays a report that is started with PC=YES.

```
DXR1001 JR21212 STOR STATS
   PC:YES  MAXCSA:N/A
CSA USE:ACNT:   0K  AHWM:   0K CUR: 4362K HWM: 5830K
   ABOVE 16M:  78 4376K   BELOW 16M:  23   32K
CLASS  TYPE SEGS  MEM  TYPE SEGS  MEM  TYPE SEGS  MEM
ACCNT  T-1    1   64K  T-2    1   64K  T-3    1    4K
PROC   WRK   11   58K  SRB   20   20K  OTH    2    2K
MISC   VAR   68 4497K  N-V    6   22K  FIX    1   24K
```

Because this is an IRLM report that results from specifying PC=YES, the storage used, shown for accountable storage, is that of IRLM private storage used for the IRLM lock control structures.

This example illustrates what can happen when an application generates a high IRLM lock contention rate. Notice that a large amount of storage used is for SRB and below the 16MB line.

Related Reading: For more information about reducing lock contention and on tuning your system, see *IBM DATABASE 2 Administration Guide* and *DB2 for OS/390 Version 5: Data Sharing: Planning and Administration*.

Example 6 for F irlmproc,STATUS

When the following command is entered on the system console:

```
F PR21PROC,STATUS,TRACE
```

The response on the console is:

```
DXR179I PR21034 TRACE USAGE
TRACE BUFFER STORAGE IN USE:   256KB
MAXIMUM NUMBER OF TRACE BUFFERS ALLOWED PER TRACE TYPE:  10
TRACE TYPE   ACTIVE   BUFFERS IN USE   CTRACE WRITER
----------   ------   --------------   -------------
   SLM          N            0              N
   XIT          Y            2              N
   XCF          N            0              N
   DBM          N            0              N
   EXP          Y            1              N
   INT          Y            1              N
```

The example shows the storage currently allocated for IRLM tracing is 256 KB, the maximum number of trace buffers allowed per trace type is set to 10, and the external CTRACE writer is not active. The trace size for each buffer is 64 KB. You can change the maximum number of trace buffers that can be used by issuing the following command: MODIFY *irlmproc*,SET,TRACE=*nnn*.

Related Reading: For more information about trace types, see Chapter 75, “TRACE CT,” on page 787

Chapter 73. STOP CQSJOB

Format

Usage

The STOP (P) cqsjobname command is used to normally shut down CQS. The command is rejected if any active IMS subsystems are currently connected to CQS.

The cqsjobname identifies the job name for the CQS to be stopped.

© Copyright IBM Corp. 1974, 2003

Chapter 74. STOP IRLMPROC

Format

Usage

The STOP (P) irlmproc command is used to normally shut down IRLM. The command is rejected if any active IMS subsystems are currently identified to IRLM.

The irlmproc identifies the procedure name for the IRLM to be stopped.

Example

z/OS SYSTEM 1 SC:

```
P IR21I
```

Response SYSTEM 1 SC:

```
DXR110I IR21001 STOP COMMAND ACCEPTED
IXL030I CONNECTOR STATISTICS FOR LOCK STRUCTURE LOCK2, 070
  CONNECTOR IRLMGRPR$IR21001:
IXL031I CONNECTOR CLEANUP FOR LOCK STRUCTURE LOCK2, 071
  CONNECTOR IRLMGRPR$IR21001, HAS COMPLETED.
DXR121I IR21001 END-OF-TASK CLEANUP SUCCESSFUL - HI-CSA    732K  -
   HI-ACCT-CSA    132K
IEF352I ADDRESS SPACE UNAVAILABLE
$HASP395 IR21I    ENDED
```

Response SYSTEM 2 SC:

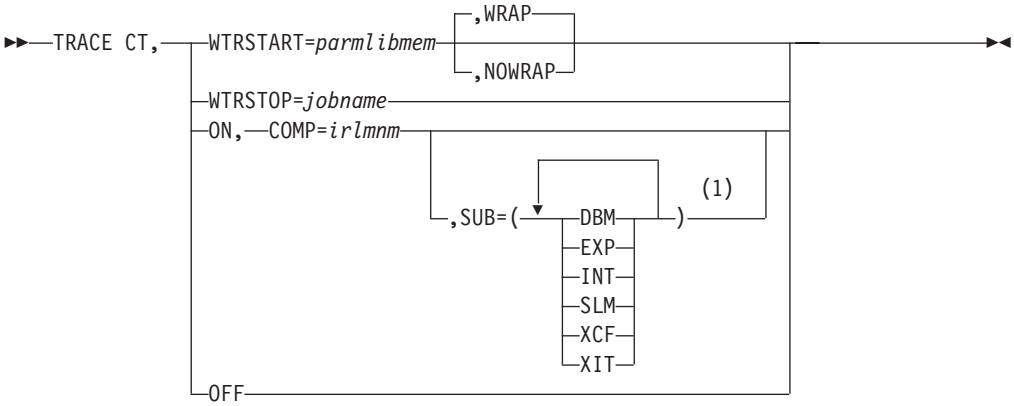
```
NONE
```

Explanation: The operator on system 1 has terminated the IRLM procedure named IR21I. No operator action on system 2 is required.

© Copyright IBM Corp. 1974, 2003

Chapter 75. TRACE CT

Format



Notes:

1 The same trace type can be specified only once.

Usage

Use the z/OS TRACE CT command to start, stop, or modify an IRLM diagnostic trace. IRLM does not support all the options available on the TRACE CT command described in *OS/390 MVS System Commands* (GC28-1781).

The impact of setting `TRACE CT ON` is that each active subname type requires up to .7 MB of ECSA. Because IRLM initializes its own traces when it starts, the `DISPLAY TRACE` command shows that all traces are off. After you issue the `TRACE ON` command, the reports are accurate except for the two subname types, INT and EXT, which cannot be turned off.

This command can be entered only from the master console. The command requires an appropriate level of z/OS authority, as described in *OS/390 MVS System Commands* (GC28-1781).

CT

Specifies the component trace (instead of the other trace options available on the z/OS TRACE command).

WTRSTART=*parmlibmem*

Identifies the name of the member that contains the source JCL that invokes the external writer and defines the data set to which external writer writes the trace buffers. The member can be a SYS1.PROCLIB cataloged procedure or a job. Many installations use a cataloged procedure in SYS1.PROCLIB.

After you enter a TRACE CT,WTRSTART command, turn the trace on and connect the writer using the WTR parameter in the reply for a TRACE CT command, as shown in “Examples” on page 789

WRAP

Specifies that when the system reaches the end of the data set or

© Copyright IBM Corp. 1974, 2003

group of data sets, it writes over the oldest data at the start of the data set or the start of the first data set in the group. The primary extents of the data set are used.

NOWRAP
Specifies that the system stops writing to the data set or data sets when they are full. The primary and secondary extents of the data sets are used.

WTRSTOP=*jobname*
Identifies the job name of a component trace external writer to be stopped. The data sets that the writer used are closed.

The job name is either:

- A member name, if the source JCL is a procedure.
- The name on a JOB statement within the source JCL.

ON
Turns on the trace.

COMP=*irlmssnm*
The IRLM subsystem name.

SUB=*subname*
Identifies the type of sublevel trace desired. This parameter can be any of the following values:

DBM
Trace interactions with the identified DBMS.

EXP
Trace any exception condition. EXP is automatically turned on when IRLM starts running and is permanently turned on.

INT
Trace member and group events outside of normal locking activity. INT is automatically turned on when IRLM starts running and is permanently turned on.

SLM
Trace interactions with the z/OS locking component.

XCF
Trace all interactions with z/OS Cross-System coupling services.

XIT
Trace just asynchronous interactions with the z/OS locking component. XIT is automatically turned on when IRLM starts running, but it can be turned off using the `z/OS TRACE CT` command.

OFF
Turns off the trace. If IRLM is connected to a component trace external writer, the system forces an implicit disconnect.

In the case of the EXP and INT sublevel traces, the OFF parameter stops the traces from writing to the external writer. However they continue to write to buffers. These traces are used by IMS for debugging purposes.

Ensure IRLM Load Module Is in z/OS Link List

Because this command uses z/OS component trace services, make sure that the IRLM start and stop load module, DXRRL183, is in the z/OS link list.

Sample External Writer Procedure

Here is a sample procedure for the IRLM external writer:

```
//CTWTR    PROC
//         EXEC PGM=ITTTRCWR
//TRCOUT01 DD   DSNAME=SYS1.WTR1,DISP=OLD
//TRCOUT02 DD   DSNAME=SYS1.WTR2,DISP=OLD
```

Displaying a Trace

To display a trace, use the z/OS DISPLAY command:

```
D TRACE,COMP=IRLM
```

Examples

Example 1 for TRACE CT

Here is an example sequence of commands to start and stop an IRLM DBM trace. In this example, the trace data is written to an external writer data set identified in procedure CTWTR:

```
TRACE CT,WTRSTART=CTWTR
TRACE CT,ON,COMP=IRLM,SUB=(DBM)
⋮
  (z/OS asks for a reply.)
⋮
R 15,WTR=CTWTR,END
TRACE CT,OFF,COMP=IRLM,SUB=(DBM)
⋮
  (Wait a while to make sure trace buffers are externalized.)
TRACE CT,WTRSTOP=CTWTR
```

Example 2 for TRACE CT

Here is a sample procedure to start and stop traces in wrap-around mode. The traces captured in this procedure are saved in a limited number of buffers that are provided by IRLM. Each buffer is reused when the previous buffer is filled.

```
TRACE CT,ON,COMP=IRLM
⋮
  (z/OS asks for a reply.)
⋮
R 15,END
TRACE CT,OFF,COMP=IRLM
```

Chapter 76. CANCEL/FORCE ODBA

The CANCEL and FORCE commands are used to terminate an ODBA application address space with an active connection to an IMS subsystem.

Format

Usage

The CANCEL jobname command is used to terminate an ODBA application address space with an active connection to an IMS subsystem. In this command, *jobname* is the ODBA application address space.

If there are no active threads under this connection to IMS, allow the CANCEL command to be processed by z/OS. If there appear to be active threads (IDTTHDCT is greater than zero), the CANCEL command is rejected and message DFS08051 is issued.

Once the CANCEL command has been issued, no additional threads will be allowed to schedule from the target address space until the connection to IMS has been successfully terminated and re-initialized.

The CANCEL is rejected to protect the IMS control region from ABENDU0113 or other related abends. If it is invalid for threads to be active at this time, gather documentation to determine the reason for the active thread(s).

Format

Usage

The FORCE jobname command is used to terminate an ODBA application address space with an active connection to an IMS subsystem if the CANCEL command was rejected. In this command, *jobname* is the ODBA application address space.

Recommendation: Attempt the CANCEL command prior to attempting the FORCE command.

If this is the first time that FORCE has been issued against the current iteration of the ODBA application address space, DFS1S100 will change the FORCE to a CANCEL and allow z/OS to process the CANCEL command.

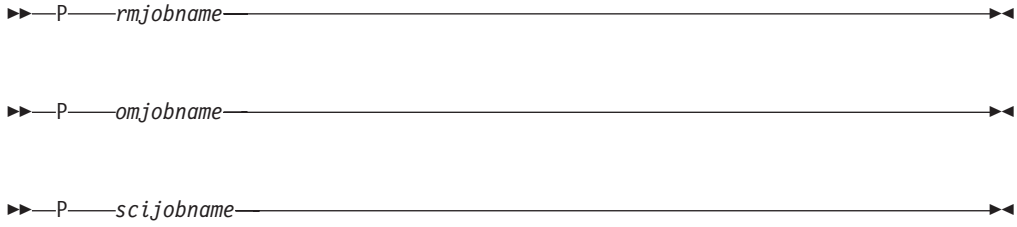
If the CANCEL command was rejected and a prior FORCE command was issued against the current iteration of the ODBA application address space and was not successful, issue a second FORCE command to allow z/OS to process the second FORCE command.

© Copyright IBM Corp. 1974, 2003

Use the FORCE command with caution. By allowing z/OS to CANCEL the address space, it is possible that the IMS control region may terminate abnormally depending on the actual state of the active thread(s). Resolve the outstanding active threads prior to issuing the FORCE command.

Chapter 77. STOP CSL Address Spaces

Format



Usage

The STOP (P) rmjobname, P omjobname, and P scijobname command is used to shut down the Resource Manager (RM), Operations Manager (OM), and Structured Call Interface (SCI) address spaces in an IMSplex.

rmjobname
Identifies the job name of the RM address space to be stopped.

omjobname
Identifies the job name of the OM address space to be stopped.

scijobname
Identifies the job name of the SCI address space to be stopped.

For more information about shutting down individual CSL address spaces, see *IMS Version 9: Common Service Layer Guide and Reference*.

To shut down the CSL as one unit, issue the CSL SHUTDOWN command to any SCI in the IMSplex with the z/OS MODIFY command interface.

Note: Before issuing the z/OS MODIFY command, issue a /CHE FREEZE command to terminate all IMSplex members that might be connected to the CSL.

To shut down a CSL on one z/OS image, issue the following z/OS MODIFY command:

```
F scijobname,SHUTDOWN CSLLCL
```

where scijobname is the name of the SCI in the CSL.

This command shuts down the CSL on the z/OS image associated with the SCI that receives the command. Use this version of the command to shut down the CSL on a single z/OS image in an orderly way.

To shut down an entire IMSplex, issue the following z/OS MODIFY command:

```
F scijobname,SHUTDOWN CSLPLEX
```

where scijobname is the name of the SCI in the CSL.

This command shuts down the CSL managers on all z/OS images in a single IMSplex associated with the SCI that receives the command.

© Copyright IBM Corp. 1974, 2003

For more information about shutting down the CSL using z/OS commands, see *IMS Version 9: Common Service Layer Guide and Reference*.

Part 4. IMS Transport Manager Subsystem Commands

Chapter 78. Introduction 797

Chapter 79. DEFINE 799
Format 799
Usage. 799

Chapter 80. DISPLAY 801
Format 801
Usage. 801
Examples 805
Example 1 for DISPLAY Command 805
Example 2 for DISPLAY Command 805
Example 3 for DISPLAY Command 805
Example 4 for DISPLAY Command 806
Example 5 for DISPLAY Command 806
Example 6 for DISPLAY Command 806

Chapter 81. SET 807
Format 807
Usage. 807

Chapter 82. START 809
Format 809
Usage. 809

Chapter 83. STOP 811
Format 811
Usage 811

© Copyright IBM Corp. 1974, 2003

Chapter 78. Introduction

Transport Manager Subsystem commands can be issued either from a SYSIN data set as an automated process during component start or from a z/OS console using the z/OS `MODIFY` command (`F procname,command`).

Transport manager commands are:

- `DEFINE`
- `DISPLAY`
- `SET`
- `START`
- `STOP`

© Copyright IBM Corp. 1974, 2003

Chapter 79. DEFINE

Format

Usage

The DEFINE command is used to define one or more transport manager subsystems that support IMS components for one or more global service groups.

A START SYSTEM(name) command is required to start the TMS once it has been defined.

SYSTEM
Specifies the first portion (1 to 5 characters) of the VTAM APPLID used by the other transport managers to which this transport manager should connect. The name specified should match the SET APPLID. A system name that matches the APPLID of this transport manager (specified on SET) causes an informational message to be issued; other system names will continue to be processed.

Transport manager will not communicate with an LU whose name has not been defined by a DEFINE SYSTEM command.

STARTED
Specifies that the system should be started automatically when the TMS is started or (if the TMS is already started) when the DEFINE command is processed.

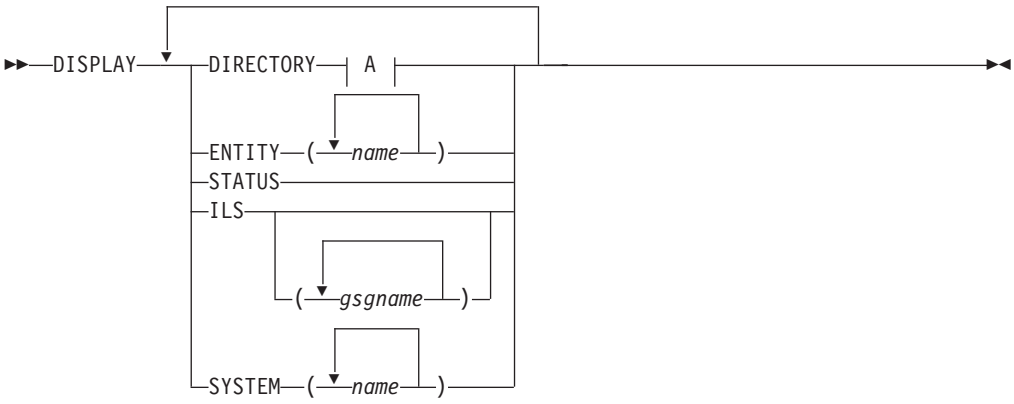
STARTED is the default.

STOPPED
Specifies that the system should not be started automatically when the TMS is started or (if the TMS is already started) when the DEFINE command is processed.

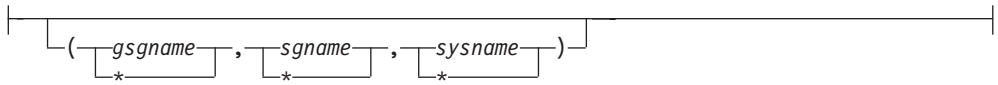
© Copyright IBM Corp. 1974, 2003

Chapter 80. DISPLAY

Format



A:



Usage

The DISPLAY command is used to provide information about the definition and status of resources related to transport manager. Transport manager only provides communication services for the active and tracking site IMSs; it has no knowledge of the activities at either site.

To see information related to active site IMS or tracking site IMS use the IMS /DISPLAY command.

DIRECTORY
Displays current entity (IMS subsystem or ILS task) information from the transport manager directory. If no entity (or list of entities) is specified, all information in the directory is displayed. The directory information includes information on the global service group (GSG) name, the service group (SG) name, the system name, the instance name, and the component name.

If you specify *, then ALL information for the entity is displayed.

ENTITY
Displays entities (IMS subsystems and ILS tasks) currently identified to this transport manager. If no list of entity names is provided, all are displayed. Entity information is displayed for the following:

MSGNAME
The IMS ID or job name of the entity. This is included in various TMS messages (prefix ELX) to identify the entity.

GSGname
The global service group name.

© Copyright IBM Corp. 1974, 2003

SGname
The service group name.

INSTANCEname
The TMS instance name.

COMPONENTname
The TMS component name.

APPLID
The VTAM APPLID assigned to the entity.

STATUS
The relationship between the entity and the transport manager. The status of the entity can be any of the following:

IDEN0
APPLID resource name is allocated for the entity during the IDENTIFY processing.

IDEN1
APPLID resource name is committed for the entity during the IDENTIFY processing.

ACT
The entity is active.

FAIL
The entity has failed during the IDENTIFY processing.

STATUS
Displays the status information of this transport manager as the following:

TRANSPORT MANAGER
The version number of the transport manager.

STATUS
The status of the entity can be any of the following:

STARTING
Transport manager start requested.

STARTED
Transport manager started.

STOPPING
Transport manager stop requested.

STOPPED
Transport manager stopped.

INSTANCE
The instance name of the transport manager.

APPLID/SYSTEM
The system name of the transport manager.

ENTITIES
The number of the active entities (including the TMS itself).

SYSTEMS
The number of the active systems.

ILS
Displays information on all or on specific global service groups for which ILS is currently started in a transport manager subsystem address space. A `DISPLAY`

ILS command will not complete while a volume mount is outstanding for other than the first volume of a multivolume log data set.

GSG.SG SUMMARY
Summary information for each tracking subsystem being supported.

GSG
Global service group name (from the START ILS command).

SG
Service group name of the tracking subsystem for which this group of display information applies.

MODE
VTAM mode name used for conversations (from the START ILS command).

MAXCONV
The maximum number of conversations allowed (from the START ILS command).

CONV
The number of conversations currently allocated between this ILS and the tracking subsystem.

Q-REQ
The number of *query* requests received from the tracking subsystem since communication began with the tracking subsystem. *Query* requests are sent periodically by the tracking subsystem to determine if active service group IMS subsystems have run without communicating with the tracking subsystem.

G-REQ
The number of *gap* requests received from the tracking subsystem since communication began with the tracking subsystem.

DS-SENT
The number of log data sets successfully sent to the tracking subsystem since communication began with the tracking subsystem. ILS actually sends contiguous portions of log data sets based on the gaps in log data at the tracking site. When all of one of these contiguous portions is sent, the value of this display field is incremented.

DATASETS
If any log data sets are currently under consideration for transport to the tracking subsystem described above, they are listed below a separator line. If there is no data set activity, there will be no separator.

STATUS
Indicates the current state of processing for a particular data set.

ACTIVE
The data set is currently being transported, no delays other than waits for data set read have been detected.

W-CONV
The data set may be selected for transmission as soon as a conversation is available.

ALLOC-IP
The data set is queued for or in the process of allocation or deallocation. This involves z/OS ENQ/DEQ (for OLDS data sets),

z/OS DYNALLOC, z/OS OPEN/CLOSE functions. Only one data set is processed through this function at a time, so a delay of one may delay others. The most likely cause of a delay in this state would be an unsatisfied tape mount.

DBRC-IP
ILS logic is awaiting completion of a DBRC request related to this data set. The most likely cause of a delay in this state would be contention for the RECON data sets.

W-SEND
The data set is currently being transported, but VTAM is not accepting data as fast as ILS is attempting to send it; send is currently waiting for completion of a previous VTAM send. This state generally reflects the results of VTAM pacing controls that limit the ILS to tracking subsystem data rate.

W-TRKPAC
The data set is currently being transported, but the tracking subsystem is not accepting data as fast as ILS is attempting to send it. This state generally reflects the tracking subsystem's inability to write log data as fast as ILS can send it.

W-TRKOK1
The data set is part of a gap; the tracking subsystem has not yet approved actual transport of data sets to fill the gap.

W-TRKOK2
ILS is waiting for the tracking subsystem to approve transport of this specific data set.

RECORDS-SENT
The number of records, in decimal, that have been sent. The number is of the form `nnnK`, where `K` represents 1024 records. The number is rounded off to the nearest K.

NAME
The IMSID of the active system that created the log data set or the job name of the BATCH or BBO job that created the log data set.

VOLSER
The volume serial number, if any, used in allocation of the data set. If this field is blank, the data set is (expected to be) cataloged.

DSNAME
The data set name that is or will be read for transport. This may be a primary or secondary copy of an OLDS or SLDS data set.

SYSTEM
Displays status information about the specified systems. The system name is specified by the `DEFINE` command and status are displayed. If no list is provided, all are displayed. The status of the system can be any of the following:

RALOC
Allocation requested for the system.

ALOC
The system is actually allocated.

RSTA
Start requested for the system.

STA
The system is actually started.

RSTO
Stop requested for the system.

STO
The system is actually stopped.

Examples

The following are examples of output from the DISPLAY command.

Example 1 for DISPLAY Command

Entry ET:

```
DISPLAY DIRECTORY
```

Response ET:

```
ELX0170I TRANSPORT MANAGER DISPLAY: 92.078   13.48.29.0
GSG      SG       SYSTEM   INSTANCE COMPONENT
*        *        TMP2     *        TMANAGER
FUNDS    SITEA    TMP2     IMSA     LOGGER
FUNDS    SITET    TMP4     IMSC     LOGROUTR
FUNDS    SITEA    TMP2     IMSB     LOGGER
FUNDS    SITEA    TMP2     *        ILSEND
```

Explanation: When the DIRECTORY object keyword is completely processed, this message is issued for current entity information from the transport manager directory.

Example 2 for DISPLAY Command

Entry ET:

```
DISPLAY DIRECTORY
```

Response ET:

```
ELX0170I TRANSPORT MANAGER DISPLAY: 92.078   13.58.39.0
*** NO DIRECTORY TO DISPLAY ***
```

Explanation: The DIRECTORY has no entities to display. This message is issued when transport manager has not started by the time the DISPLAY command is processed.

Example 3 for DISPLAY Command

Entry ET:

```
DISPLAY ENTITY
```

Response ET:

```
ELX0170I TRANSPORT MANAGER DISPLAY: 92.090   13.48.29.0
MSGNAME  GSG      SG       INSTANCE COMP     APPLID   STATUS
TMP      *        *        *        TMANAGER TMP2001  IDEN1,ACT
TMPA     FUNDS    SITEA    IMSA     LOGGER   *        IDEN0
```

Explanation: When the ENTITY object keyword is completely processed, this message is issued to display entities currently identified to this transport manager.

Example 4 for DISPLAY Command

Entry ET:

```
DISPLAY SYSTEM
```

Response ET:

```
ELX0170I TRANSPORT MANAGER DISPLAY: 92.078  13.54.51.1
SYSTEM   STATUS
TMP4     ALOC,STA
TMP3     ALOC,STA
TMP5     ALOC,STA
```

Explanation: When the SYSTEM object keyword is completely processed, this message is issued to display status information for the specified systems.

Example 5 for DISPLAY Command

Entry ET:

```
DISPLAY ILS
```

Response ET:

```
ELX0170I TRANSPORT MANAGER DISPLAY: 91.105  12.07.24.7
GSG: IMSGSG1  SG: STLSITE1 MODE: ILSMODE  MAXCONV:  2
CONV:  2      Q-REQ:    92 G-REQ:     3   DS-SENT:     5
**** DATASETS ****************************************************************
STATUS   RECORDS-SENT NAME     VOLSER DSNAME
W-SEND            12K IMSA     L00813 BURKES.PAYROLL.SLDSP1
ACTIVE            72K IMSA            KENTT.PAYROLL.DFSOLP03
W-CONV             0K IMSA     L00803 BURKES.PAYROLL.SLDSP2
```

Explanation: When the ILS object keyword is completely processed, this message is issued to display isolated log sender status for the specified GSG name.

Example 6 for DISPLAY Command

Entry ET:

```
DISPLAY STATUS
```

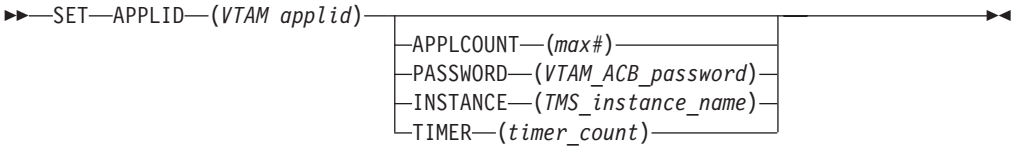
Response ET:

```
ELX0170I TRANSPORT MANAGER DISPLAY: 90.078   13.56.59.5
TRANSPORT MANAGER V5 R0     STATUS:  STARTING,STARTED
INSTANCE:  ELX     APPLID/SYSTEM:  TMP2
ENTITIES:  1       SYSTEMS:  3
```

Explanation: When the /DISPLAY command is entered with the STATUS object keyword, this message is issued to display the status of resources related to transport manager.

Chapter 81. SET

Format



Usage

The /SET command allows you to specify various parameters that typically stay in effect for the duration of an execution of the transport manager task.

APPLID
Specifies a 1- to 5-character name that becomes the first portion of the series of VTAM APPLIDs used by transport manager and related IMS subsystems. This name also becomes the “system name” of the CPC running this transport manager (used in the naming scheme used by transport manager and users of transport manager).

The name used for the APPLID must match the first portion of the name (specified on the DEFINE SYSTEM command) or the ACBNAME=acbname used for a series of VTAM APPL definition statements. See the *IMS Version 9: Installation Volume 2: System Definition and Tailoring*.

When transport manager is connected to VTAM (that is, the ACB is open), the APPLID keyword is rejected. To avoid definitional conflicts, SET APPLID must be specified before DEFINE SYSTEM commands are issued.

APPLCOUNT
Specifies the number of VTAM applids that are to be defined to VTAM for use by this transport manager. The number must exceed by at least 1 the maximum number of IMS subsystems ever expected concurrently to execute on this CPC plus the maximum instances of isolated log sender to be started for this transport manager.

PASSWORD
Specifies the VTAM ACB password to be used for the transport manager. This specification is tied to the VTAM APPL PRTCT=password definition. A single password is used for all APPL names.

INSTANCE
Specifies a 1 to 4 character name that becomes the instance name (subsystem id) of this execution of transport manager. This value, combined with the TMI name specified in the DFSRSRxx PROCLIB member, the IMSCTRL macro, the DLIBATCH procedure or the DBBBATCH procedure, allows you to use multiple instances of the transport manager within a single system. With different instances of transport manager, you can run test and production subsystems within the same IMS system.

The default value for the instance name is ELX.

Once the TMS is started, the INSTANCE keyword will be rejected.

TIMER
Specifies the interval, in seconds, to be used for automatic restart of

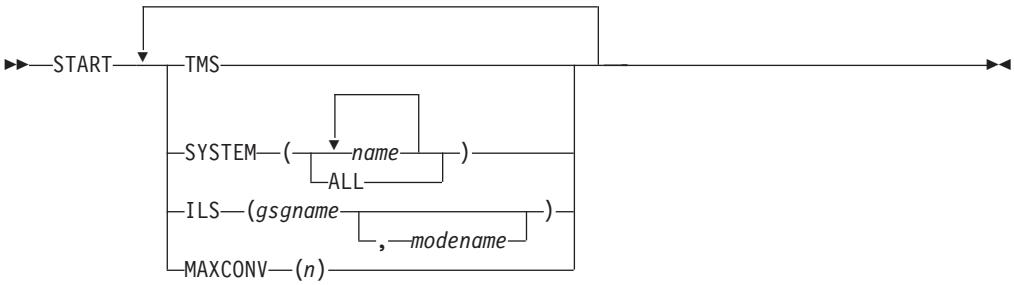
© Copyright IBM Corp. 1974, 2003

transport-manager-to-transport-manager conversations. Automatic restart attempts are made after conversation failures and are continued until a conversation is established or a `STOP SYSTEM` or `STOP RETRY` command is issued.

The default value is 300 seconds.

Chapter 82. START

Format



Usage

The /START command is used to activate the objects specified (TMS, SYSTEM, or ILS).

TMS

Specifies that the transport manager is to be started. The START TMS command should be issued after the initial SET and DEFINE commands.

SYSTEM

Specifies that conversations to the specified systems are to be started. This command may be required after a network outage where the transport managers are up but there has been no network connectivity. Specifying SYSTEM(ALL) causes start to be attempted for all systems not already started or starting.

ILS

Specifies that an isolated log sender task is to be started.

gsgname specifies the 1- to 8-character name of the global service group to be serviced by this instance of isolated log sender.

modename specifies the VTAM mode name to be used when allocating conversations for a transfer request. If modename is not specified, the TMS default mode name, TMDEFLT, is used.

MAXCONV

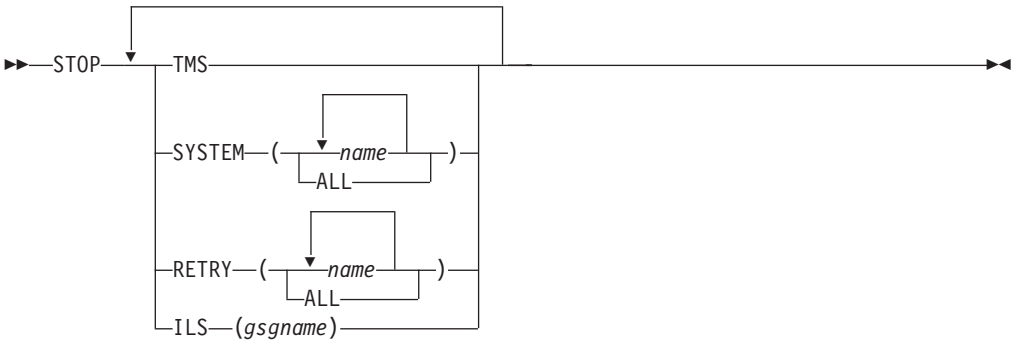
Specifies, for an ILS task, the maximum number of concurrent data set transfers to be used to process one transfer request. The most efficient value is highly dependent on network capacity and routing, as well as on disk configurations and allocations on both local and remote sites.

The default value is 2. The maximum value is 30.

© Copyright IBM Corp. 1974, 2003

Chapter 83. STOP

Format



Usage

The STOP command is used to stop the objects specified. In order to stop the entire transport manager subsystem, you can use z/OS STOP command (P procname) from the z/OS console.

TMS
Specifies that the entire TMS subsystem is to be stopped.

SYSTEM
Specifies that conversations to the specified systems are to be stopped. Requests from other systems to establish conversations will be rejected.

RETRY
Specifies that automatic conversation allocation retry is to be stopped for the specified systems or for ALL systems. Requests from other systems to establish conversations will be accepted.

Automatic conversation allocation retry is allowed again after a conversation is successfully established with the specified systems.

ILS
Specifies that an isolated log sender task is to be stopped.

gsgname specifies the 1- to 8-character name of the global service group being serviced by the instance of isolated log sender to be stopped.

© Copyright IBM Corp. 1974, 2003

Part 5. Base Primitive Environment Commands

Chapter 84. BPE Commands 815
BPE Command Syntax and Invocation 815
BPE Command Invocation 815
BPE Wildcard Character Support 816
Specifying IMS Component Command Parameters 816
BPE TRACETABLE Commands 817
Format of DISPLAY TRACETABLE Command 817
Usage of DISPLAY TRACETABLE Command 817
Format of UPDATE TRACETABLE Command 821
Usage of UPDATE TRACETABLE Command 822
BPE DISPLAY VERSION Command 824
Format of DISPLAY VERSION Command 824
Usage of DISPLAY VERSION Command 824
BPE USEREXIT Commands 825

© Copyright IBM Corp. 1974, 2003

Chapter 84. BPE Commands

BPE Command Syntax and Invocation

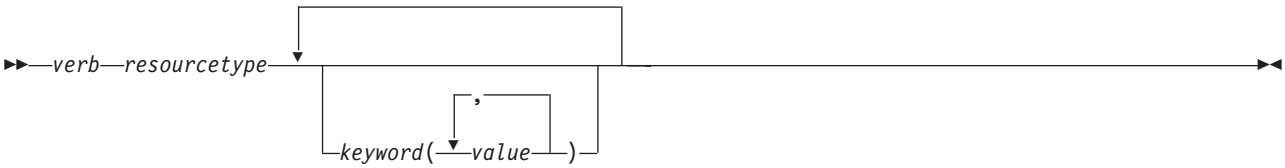
BPE supports two command formats: a verb only format, and a verb-resourcetype format.

The verb only format consists of a verb, followed by zero or more keyword-value pairs, with the values enclosed in parentheses.

BPE Verb Only Command Syntax

The verb-resourcetype format consists of a verb, a resource type, and zero or more keyword value pairs.

BPE Verb-Resource Type Command Syntax



verb
A command verb representing an action. Some verb examples are DISPLAY, UPDATE, and REFRESH.

resourcetype
The type of resource that is operated on by the verb. Some resource examples are TRACETABLE and USEREXIT.

keyword(value)
A set of zero or more keywords and values that represent attributes, filters, or other modifiers that apply to the command. For example, NAME() to identify the specific resource(s) or LEVEL() to specify a trace level.

BPE Command Invocation

You can only invoke BPE commands through the MVS MODIFY command. The following diagram illustrates the general syntax for entering commands through the modify interface.

BPE Command Invocation

© Copyright IBM Corp. 1974, 2003

F The MVS modify command.

jobname
The jobname of the address space to which the command is directed.

command
The command being issued.

BPE Wildcard Character Support

Some parameters on BPE commands support wildcard characters for pattern matching. For such parameters, you can use the following wildcard characters:

***** Matches zero or more characters

% Matches exactly one character

The following examples illustrate some uses of wildcard characters.

BE* Matches any string beginning with ″BE″, of any length. For instance: BE, BEE, BEEBLEBROX.

%%S Matches any three-character string ending with an ″S″. For instance: IMS, CQS.

R*S*T%R
Matches any string beginning and ending wtih ″R″, having an ″S″, followed by a ″T″ in the middle, with any number of intervening characters between the first ″R″, the ″S″, and the ″T″, and exactly one character between the ″T″ and the final ″R″. For instance: ROASTER, ROSTER, RESORTER, RESCEPTOR, RSTZR.

***** Matches any string.

Specifying IMS Component Command Parameters

BPE commands enable you to display and update resources that BPE manages. Some resource types are defined and owned by BPE itself. These resource types are known as ″system resource types.″ Commands that specify system resource types can be issued to any IMS component running in a BPE environment. For example, BPE defines several BPE system trace table types like DISP, STG, and CBS. These trace tables exist in every BPE address space. Commands to display and update these trace table types can be issued to any BPE address space.

Other resource types are defined and owned by the IMS component that is using BPE services. These resource types are known as ″component resource types″ or ″user-product resource types.″ Commands that specify component resource types can only be issued to the IMS component that defines those types. For example, CQS defines several CQS-specific trace tables such as STR, CQS, and INTF. Commands to display and update these trace table types can be issued only to CQS address spaces.

BPE commands also provide the ability to restrict the resource types upon which a command operates to either those owned by BPE, or to those owned by the IMS component of the address space to which the command is issued. This is done through the OWNER keyword on commands that support OWNER. Use `OWNER(BPE)` to restrict the command operation to resource types that BPE owns and defines (system resource types). Use `OWNER(`*`component_type`*`)` to restrict the command operation to resource types that the IMS component address defines and owns (component resource types). Table 190 on page 817 shows the valid values for the OWNER parameter, and the address space types to which they apply:

Table 190. Valid Values for OWNER Parameter

OWNER	Address Space Type
BPE	Any IMS component running in a BPE address space
CQS	Common Queue Server
OM	Operations Manager
RM	Resource Manager
SCI	Structured Call Interface

BPE TRACETABLE Commands

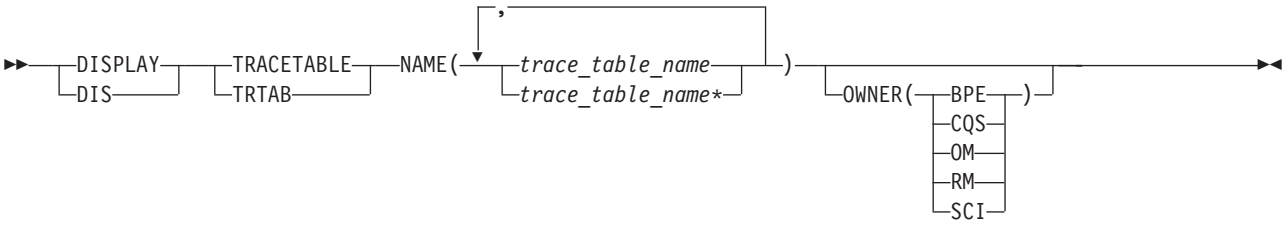
The TRACETABLE resource type refers to the internal BPE-managed trace tables defined either by BPE (for example: DISP, CBS, STG, LATC), or by the IMS component using BPE (for example: CQS, OM, RM, SCI). Two command verbs operate on the TRACETABLE resource type:

DISPLAY Display trace level and number of trace table pages of specified trace tables.

UPDATE Update trace level attribute of specified trace tables.

Format of DISPLAY TRACETABLE Command

Use this command to display the current attribute settings for the requested trace tables.



Usage of DISPLAY TRACETABLE Command

DISPLAY | DIS

A required parameter, which specifies that the action against the specified resource is to display attributes of the resource.

TRACETABLE | TRTAB

A required parameter, which specifies that the resource type being acted upon is a BPE-managed trace table.

NAME(*trace_table_name***)**

A required parameter, which specifies the name of the trace table type or types about which you want attributes displayed. You can specify a single trace table name or a list of trace table names separated by commas. Trace table names can contain wildcard characters. See “BPE Wildcard Character Support” on page 816 for more information about using wildcard characters. Trace table names can be BPE-defined trace tables or IMS component-defined trace tables.

You can display BPE-defined trace tables for any IMS component address space that is using BPE. These BPE-defined trace table types are available:

AWE Asynchronous work element (AWE) trace table

CBS Control block services trace table

CMD Command trace table

DISP Dispatcher trace table

HASH Hash trace table

ERR BPE Error trace table

LATC Latch trace table

MISC Miscellaneous trace table that is used only by IMS Service for trap traces

SSRV System services trace table

STG Storage service trace table

USRX User exit routine trace table

You can display CQS-defined trace tables only for CQS address spaces. These CQS-defined trace table types are available:

CQS CQS trace table

ERR CQS error trace table

INTF CQS interface trace table

STR CQS structure trace table

You can display OM-defined trace tables only for OM address spaces. These OM-defined trace table types are available:

CSL Common Service Layer (CSL) trace table

ERR OM error trace table

OM Operations Manager (OM) processes trace table

PLEX IMSplex trace table for OM processing for a specific IMSplex

You can display RM-defined trace tables only for RM address spaces. These RM-defined trace table types are available:

CSL Common Service Layer (CSL) trace table

ERR RM error trace table

PLEX IMSplex trace table for RM processing for a specific IMSplex

RM Resource Manager (RM) processes trace table

You can display SCI-defined trace tables only for SCI address spaces. These SCI-defined trace table types are available:

CSL Common Service Layer (CSL) trace table

ERPL SCI Error Parameter List trace table

ERR SCI error trace table

INTF SCI interface trace table

INTP SCI interface parameter trace table

PLEX IMSplex trace table for SCI processing for a specific IMSplex

SCI Structured Call Interface (SCI) processes trace table

OWNER(BPE | CQS | OM | RM | SCI)
An optional parameter that specifies the owner of the trace table type or types about which you want attributes displayed. You can specify one of the following values:

BPE For all IMS components that are running in a BPE address space.

CQS For CQS address spaces only

OM For OM address spaces only

RM For RM address spaces only

SCI For SCI address spaces only

The OWNER parameter acts as a filter to help you select which trace tables you want to display. For example, you could specify `NAME(*) OWNER(CQS)` to display all of the CQS-defined trace table types (CQS, ERR, STR, and INTF) in a CQS address space. You could specify `NAME(*) OWNER(BPE)` to display all of the BPE-defined trace table types in *any* BPE-managed address space. If `OWNER` is omitted, then both BPE and component trace tables might be displayed (depending on the tables specified on `NAME`).

DISPLAY TRACETABLE Command Output

The DISPLAY TRACETABLE command output consists of a header line, one line per selected trace table, and one message BPE0032I line indicating that the command has completed. Here is an example.

```
BPE0030I TABLE  OWNER  LEVEL  #PAGES
BPE0000I DISP   BPE    HIGH       12
BPE0000I STR    CQS    MEDIUM      8
BPE0032I DISPLAY TRACETABLE COMMAND COMPLETED
```

These columns are in the DISPLAY TRACETABLE output:

TABLE Specifies the name of the trace table type about which information is being displayed on the current row. Either BPE or the product using BPE owns this trace table.

OWNER Specifies the IMS component that owns the trace table. BPE-owned trace tables are system trace tables, and exist in all IMS component address spaces that use BPE. Trace tables that are specific to an IMS component show the 1- to 4-character component identifier in this column.

LEVEL Specifies the current level setting of the trace table. A trace table's level determines the volume of trace data collected. These levels are possible:

NONE No tracing is being done into the table.

ERROR Only traces for error or exception conditions are being made into the table.

LOW Only major event trace entries are made into the table.

MEDIUM Major event trace entries and some minor event trace entries are made into the table.

HIGH All trace entries are made into the table.

INACTV The trace table is inactive and cannot be used. This

status occurs only when BPE was unable to get any storage for the trace table. No tracing will be done for the indicated table type, and you cannot change the level for the trace table with the UPDATE TRACETABLE command. You must restart the address space in order to use the trace table again.

#PAGES Specifies the number of 4K (4096 byte) pages allocated for the trace table type.

Command Example 1

Display the status of the BPE dispatcher trace table (DISP).

Command:

```
F CQS1,DISPLAY TRACETABLE NAME(DISP)
```

Output:

```
BPE0030I TABLE  OWNER  LEVEL  #PAGES
BPE0000I DISP   BPE    HIGH       12
BPE0032I DISPLAY TRACETABLE COMMAND COMPLETED
```

Command Example 2

Display the status of all CQS traces.

Command:

```
F CQS1,DIS TRTAB NAME(*) OWNER(CQS)
```

Output:

```
BPE0030I TABLE  OWNER  LEVEL  #PAGES
BPE0000I CQS    CQS    MEDIUM      4
BPE0000I ERR    CQS    HIGH        4
BPE0000I INTF   CQS    LOW         8
BPE0000I STR    CQS    HIGH        8
BPE0032I DISPLAY TRACETABLE COMMAND COMPLETED
```

Command Example 3

Display the status of all traces in an SCI address space.

Command:

```
F SCI,DIS TRTAB NAME(*)
```

Output:

```
BPE0030I TABLE  OWNER  LEVEL  #PAGES
BPE0000I AWE    BPE    HIGH       6
BPE0000I CBS    BPE    HIGH       6
BPE0000I CMD    BPE    HIGH       2
BPE0000I CSL    SCI    HIGH       8
BPE0000I DISP   BPE    HIGH       8
BPE0000I ERPL   SCI    HIGH       8
BPE0000I ERR    BPE    HIGH       2
BPE0000I ERR    SCI    HIGH       4
BPE0000I HASH   BPE    HIGH       8
BPE0000I INTF   SCI    HIGH       8
BPE0000I INTP   SCI    HIGH       16
BPE0000I LATC   BPE    HIGH       8
BPE0000I MISC   BPE    HIGH       1
BPE0000I PLEX   SCI    HIGH       8
BPE0000I SCI    SCI    HIGH       8
BPE0000I SSRV   BPE    HIGH       4
```

```
BPE0000I STG    BPE    HIGH       8
BPE0000I USRX   BPE    HIGH       4
BPE0032I DISPLAY TRACETABLE COMMAND COMPLETED
```

Command Example 4

Display the status of all OM traces.

Command:

```
F OM,DIS TRTAB NAME(*) OWNER(OM)
```

Output:

```
BPE0030I TABLE  OWNER  LEVEL  #PAGES
BPE0000I CSL    OM     HIGH       4
BPE0000I ERR    OM     HIGH       4
BPE0000I OM     OM     HIGH       4
BPE0000I PLEX   OM     HIGH       8
BPE0032I DISPLAY TRACETABLE COMMAND COMPLETED
```

Command Example 5

Display the status of the PLEX trace and all traces beginning with ″C″ in the RM address space.

Command:

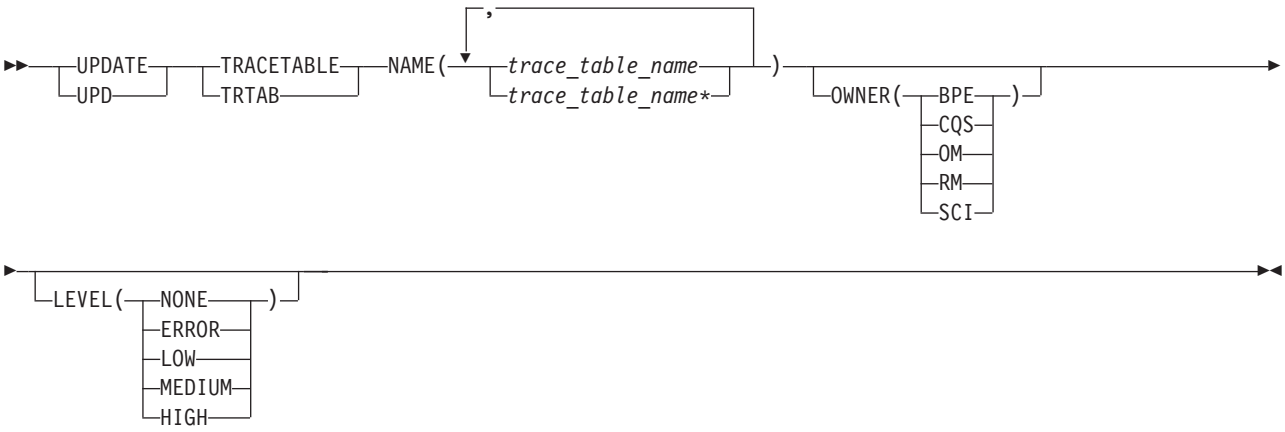
```
F RM,DIS TRTAB NAME(PLEX,C*)
```

Output:

```
BPE0030I TABLE  OWNER  LEVEL  #PAGES
BPE0000I CBS    BPE    HIGH       6
BPE0000I CMD    BPE    HIGH       2
BPE0000I CSL    RM     HIGH       4
BPE0000I PLEX   RM     HIGH       8
BPE0032I DISPLAY TRACETABLE COMMAND COMPLETED
```

Format of UPDATE TRACETABLE Command

Use this command to change the trace level setting for the requested trace tables.



Usage of UPDATE TRACETABLE Command

UPDATE | UPD
A required parameter, which specifies that the action against the trace table is to update its attributes.

TRACETABLE | TRTAB
A required parameter, which specifies that the resource type being acted upon is a BPE-managed trace table.

NAME(*trace_table_name***)**
A required parameter, which specifies the name of the trace table type or types that you want to update. You can specify a single trace table name or a list of trace table names separated by commas. Trace table names can contain wildcard characters. See “BPE Wildcard Character Support” on page 816 for more information about using wildcard characters. Trace table names can be BPE-defined trace tables or IMS component-defined trace tables.

You can update BPE-defined trace tables for any IMS component address space that is using BPE. These BPE-defined trace table types are available:

AWE Asynchronous work element (AWE) trace table

CBS Control block services trace table

CMD Command trace table

DISP Dispatcher trace table

ERR BPE Error trace table

HASH Hash trace table

LATC Latch trace table

MISC Miscellaneous trace table that is used only by IMS Service for trap traces

SSRV System services trace table

STG Storage service trace table

USRX User exit routine trace table

You can update CQS-defined trace tables only for CQS address spaces. These CQS-defined trace table types are available:

CQS CQS trace table

ERR CQS error trace table

INTF CQS interface trace table

STR CQS structure trace table

You can update OM-defined trace tables only for OM address spaces. These OM-defined trace table types are available:

CSL Common Service Layer (CSL) trace table

ERR OM error trace table

OM Operations Manager (OM) processes trace table

PLEX IMSplex trace table for OM processing for a specific IMSplex

You can update RM-defined trace tables only for RM address spaces. These RM-defined trace table types are available:

CSL Common Service Layer (CSL) trace table

ERR RM error trace table

PLEX IMSplex trace table for RM processing for a specific IMSplex

RM Resource Manager (RM) processes trace table

You can update SCI-defined trace tables only for SCI address spaces. These SCI-defined trace table types are available:

CSL Common Service Layer (CSL) trace table

ERPL SCI Error Parameter List trace table

ERR SCI error trace table

INTF SCI interface trace table

INTP SCI interface parameter trace table

PLEX IMSplex trace table for SCI processing for a specific IMSplex

SCI Structured Call Interface (SCI) processes trace table

OWNER(BPE | CQS | OM | RM | SCI)

An optional parameter that specifies the owner of the trace table type or types that you want to update. You can specify one of the following values:

BPE For all IMS components that are running in a BPE address space

CQS For CQS address spaces only

OM For OM address spaces only

RM For RM address spaces only

SCI For SCI address spaces only

The OWNER parameter acts as a filter to help you select which trace tables you want to update. For example, you could specify `NAME(*) OWNER(CQS)` to update all of the CQS-defined trace table types (CQS, ERR, STR, and INTF) in a CQS address space. You could specify `NAME(*) OWNER(BPE)` to update all of the BPE-defined trace table types in *any* BPE-managed address space. If `OWNER` is omitted, then both BPE and component trace tables might be updated (depending on the tables specified on `NAME`).

LEVEL(*level***)**

An optional parameter that sets the new tracing level for the specified trace tables. If LEVEL is omitted, the level of the specified trace tables is not changed. These levels are possible:

NONE No tracing is being done into the table.

ERROR Only traces for error or exception conditions are being made into the table.

LOW Only major event trace entries are made into the table.

MEDIUM Major event trace entries and some minor event trace entries are made into the table.

HIGH All trace entries are made into the table.

Important: You cannot change the level for the trace table type ERR. BPE forces the level to HIGH to ensure that error diagnostics are captured. Any level that you specify for the ERR trace table is ignored.

UPDATE TRACETABLE Command Output

The UPDATE TRACETABLE command output consists of message BPE0032I indicating that the command has completed:

```
BPE0032I UPDATE TRACETABLE COMMAND COMPLETED
```

Command Example 1

Update the level of the BPE dispatcher trace table (DISP) to HIGH.

Command:

```
F CQS1,UPDATE TRACETABLE NAME(DISP) LEVEL(HIGH)
```

Output:

```
BPE0032I UPDATE TRACETABLE COMMAND COMPLETED
```

Command Example 2

Update the level of all SCI trace tables to MEDIUM.

Important: You cannot change the level for the trace table type ERR -- even when using a wildcard character to select all tables with a given owner, as in this example. BPE forces the level to HIGH to ensure that error diagnostics are captured.

Command:

```
 F SCI,UPD TRTAB NAME(*) OWNER(SCI) LEVEL(MEDIUM)
```

Output:

```
BPE0032I UPDATE TRACETABLE COMMAND COMPLETED
```

BPE DISPLAY VERSION Command

Use this command to display both the version of the IMS component that is using BPE, and the version of the BPE in use.

Format of DISPLAY VERSION Command

Usage of DISPLAY VERSION Command

DISPLAY | DIS

A required parameter, which specifies that the action against the specified resource is to display attributes of the resource.

VERSION | VER

A required parameter, which specifies that the resource types being acted upon are the version number of the IMS component **and** the BPE in the current address space.

DISPLAY VERSION Command Output

The DISPLAY VERSION command output consists of a single display output line in the format **BPE00001** *comp* **VERSION**=*cv.cr.cp* **BPE VERSION**=*bv.br.bp*.

- *comp* is the IMS component ID for the address space. It is one to four characters long and can have one of the following values:
  - CQS (Common Queue Server)
  - OM (Operations Manager)
  - RM (Resource Manager)
  - SCI (Structured Call Interface)
- *cv.cr.cp* is the full version number of the IMS component, where *cv* is the version, *cr* is the release, and *cp* is the point release. Similarly, *bv.br.bp* indicates the full version number of the BPE running in the address space

Command Example 1

Display the version of a CQS address space.

Command:

```
F CQS1,DISPLAY VERSION
```

Output:

```
BPE00001 CQS VERSION = 1.3.0  BPE VERSION = 1.4.0
```

Command Example 2

Display the version of an RM address space.

Command:

```
F RM1,DISPLAY VERSION
```

Output:

```
BPE00001 RM VERSION = 1.1.0  BPE VERSION = 1.4.0
```

BPE USEREXIT Commands

Note: Throughout this section, the term “user exit routine” means “user-supplied exit routine.”

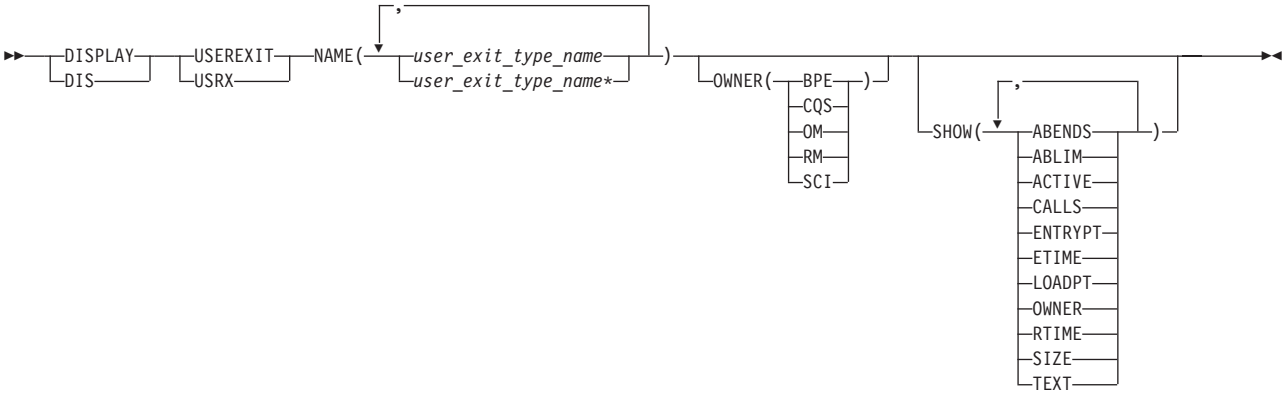
The USEREXIT resource type refers to the user exit types defined to and managed by either BPE or the IMS component using BPE (for example, CQS).

DISPLAY Display attributes of specified user exit types.

REFRESH Load new copies of the user exit modules for specified user exit types.

Format of DISPLAY USEREXIT Command

Use this command to display attributes for all modules associated with the specified user exit types.



Usage of DISPLAY USEREXIT Command

DISPLAY | DIS

A required parameter, which specifies that the action against the specified resource(s) is to display attributes of the resource(s).

USEREXIT | USRX

A required parameter, which specifies that the resource type being acted upon is a BPE-managed user exit type.

NAME(*user_exit_type_name***)**

A required parameter, which specifies the name of the user exit type or types about which you want attributes displayed. You can specify a single user exit type name or a list of user exit type names separated by commas. User exit type names can contain wildcard characters.

Related Reading:

For more information about using wildcards, see “BPE Wildcard Character Support” on page 816.

Important: The name(s) specified in this parameter are the name(s) of user exit types, *not* the name(s) of individual user exit modules.

BPE and each address space that can use BPE have different user exit types. BPE’s user exit types, as specified by OWNER(BPE), include the following:

INITTERM Initialization-Termination user exit

STATS BPE system functions statistics user exit

User exit types are defined in all CQS address spaces, as specified by OWNER(CQS), and include the following:

CLNTCONN Client Connection user exit

INITTERM Initialization-Termination user exit

OVERFLOW Queue Overflow user exit

STRSTAT Structure statistics user exit

STREVENT Structure event user exit

User exit types are defined in all OM address spaces, as specified by OWNER(OM), and include the following:

CLNTCONN Client Connection command registration and deregistration user exit

INITTERM Initialization-Termination user exit

INPUT Command input user exit

OUTPUT Output user exit

SECURITY Security checking user exit

User exit types are defined in all RM address spaces, as specified by OWNER(RM), and include the following:

CLNTCONN Client Connection and Disconnection user exit

INITTERM Initialization-Termination user exit

User exit types are defined in all SCI address spaces, as specified by OWNER(SCI), and include the following:

CLNTCONN Client Connection and Disconnection user exit

INITTERM Initialization-Termination user exit

Related Reading:

- See *IMS Version 9: Base Primitive Environment Guide and Reference* for more information about the BPE user exit routine types.
- See *IMS Version 9: Common Queue Server Guide and Reference* for more information about the CQS user exit routine types.
- See *IMS Version 9: Common Service Layer Guide and Reference* for more information about the OM, RM, and SCI user exit routine types.

OWNER(BPE | CQS | OM | RM | SCI)

An optional parameter that specifies the owner of the user exit type or types about which you want attributes displayed. You can specify one of the following values:

BPE
For all IMS components that are running in a BPE address space

CQS
For CQS address spaces only

OM
For OM address spaces only

RM
For RM address spaces only

SCI
For SCI address spaces only

The OWNER parameter acts as a filter to help you select the user exit types that you want to display. For example, you could specify `NAME(*) OWNER(CQS)` to display all of the CQS-defined user exit types in a CQS address space. If `OWNER` is omitted, then both BPE and component user exits can be displayed (depending on the exits specified on `NAME`).

SHOW(*attribute*)

An optional parameter that specifies the attributes you want to display about the requested user exits.

When you display information about user exits, each row of display output contains the requested attributes for one user exit module, in columns. Every display for user exits contains the columns labeled EXITTYPE (the type of the exit), and MODULE (the load module name of the exit). Additionally, any of the following attributes can be requested by using the SHOW parameter:

ABENDS The number of abends that have occurred in the user exit module since the last user exit refresh of that module (or since address space initialization if no refreshes have been done). BPE keeps track of the number of abends that have occurred in each user exit module. When this number reaches the number defined on the ABLIM= parameter of the EXITDEF statement for the exit's type, BPE stops calling the module. If the user exit module is refreshed, this count is reset to zero, and BPE calls the module again.

If the abend limit (ABLIM) value is not zero, and if the number of abends is greater than or equal to the abend limit value, then the user exit has reached its abend limit, and is no longer being called by BPE.

The maximum value that can be displayed in this field is 2147483647 (2^{31}-1). If the abend count exceeds this value, 2147483647 is displayed.

ABLIM The abend limit count for the user exit type, as specified on the ABLIM= parameter on the EXITDEF statement for the user exit type in the BPE exit list PROCLIB member. This is the number of times the user exit module is allowed to abend before BPE stops calling the user exit. A value of 0 indicates that there is no abend limit.

The maximum value that can be displayed in this field is 2147483647 (2^{31}-1). If the abend limit count exceeds this value, 2147483647 is displayed.

ACTIVE The number of currently active instances of the user exit. This is a point-in-time number that represents the number of calls to the user exit that have not yet returned.

The maximum value that can be displayed in this field is 999999. If the active count exceeds this value, 999999 is displayed.

CALLS The number of calls to the user exit since the last user exit refresh.

For performance reasons, serialization is not obtained when BPE collects this number. For an exit type that can run multiple instances in parallel, this number should be considered an approximation only.

The maximum value that can be displayed in this field is 2147483647 (2^{31}-1). If the call count exceeds this value, 2147483647 is displayed.

ENTRYPT The entry point address of the user exit module.

ETIME The total (cumulative) elapsed time spent in the exit module since it was last refreshed, in milliseconds.

For performance reasons, serialization is not obtained when BPE collects this number. For an exit type that can run multiple instances in parallel, this number should be considered an approximation only.

The maximum value that can be displayed in this field is 2147483647 (2^{31}-1). If the elapsed number of milliseconds exceeds this value, 2147483647 is displayed.

LOADPT The load point address of the user exit module.

OWNER The IMS component that owns the user exit type. BPE-owned user exit types are system exit types, and exist in all IMS component address spaces that use BPE. User exit types that are specific to the component show the 1- to 4-character component identifier in this column (for example, CQS).

RTIME The refresh time of the user exit module. This is the local date and time that the user exit module was last refreshed (or initially loaded, if no refreshes have been done). The format of this output field is:

```
yyyy-mm-dd hh:mm:ss.th
```

SIZE The size of the user exit load module, in bytes (displayed in hexadecimal).

TEXT 27 bytes starting from offset +04 from the module's entry point, translated to EBCDIC, with non-printable characters replaced by periods (.). This is a common location for module identification information. If your user exits contain printable identification data at this point in the module, the TEXT option enables that information to be displayed.

If the SHOW parameter is not specified, the default attributes displayed after the EXITTYPE and MODULE are OWNER, ACTIVE, and ABENDS.

The order in which you list the attributes on the SHOW parameter has no effect on the order the attributes are displayed. BPE determines the order of the attribute columns in the display output. This order is as follows:

- OWNER
- ACTIVE
- ABENDS
- ABLIM
- CALLS
- ETIME
- RTIME
- ENTRYPT
- LOADPT
- SIZE
- TEXT

Important: It is possible to request so many attributes that the length of the output line is too long to display with a WTO. If this happens, the command is processed, but some lines might be truncated. The maximum line length that BPE displays is 126 characters.

DISPLAY USEREXIT Command Output: The DISPLAY USEREXIT command output consists of a header line, one line per user exit module about which information is being displayed, and one message, BPE0032I line indicating the command has completed. For example, the command

```
F CQS1,DISPLAY USEREXIT NAME(INITTERM,STRSTAT)
```

displays the following:

```
BPE0030I EXITTYPE MODULE   OWNER    ACTIVE     ABENDS
BPE0000I INITTERM MYINIT00 CQS           0          0
BPE0000I INITTERM ZZZINIT0 CQS           0          0
BPE0000I STRSTAT  MYSTAT00 CQS           1          2
BPE0032I DISPLAY USEREXIT COMMAND COMPLETED
```

The EXITTYPE and MODULE columns are present for all DISPLAY USEREXIT commands, regardless of what is specified on SHOW. When multiple exit modules are listed for a single user exit type, the order in which they are listed is the order in which they are called.

Command Example 1: Display the status of the CQS structure event user exit type:

Command:

```
F CQS1,DISPLAY USEREXIT NAME(STREVENT)
```

Output:

```
BPE0030I EXITTYPE MODULE   OWNER    ACTIVE     ABENDS
BPE0000I STREVENT STREVX00 CQS           1          0
BPE0000I STREVENT ZZZSTEV0 CQS           0          0
BPE0032I DISPLAY USEREXIT COMMAND COMPLETED
```

In this example, there are two structure event exit modules defined that are called for CQS structure events. STREVX00 is called first, followed by ZZZSTEV0.

Command Example 2: Display the number of calls to, the elapsed time spent in, and the abend limit for all CQS user exit types:

Command:

```
F CQS1,DIS USRX NAME(*) OWNER(CQS) SHOW(CALLS,ETIME,ABLIM)
```

Output:

```
BPE0030I EXITTYPE MODULE        ABLIM      CALLS      ETIME
BPE0000I CLNTCONN CLCONX00          0          2         12
BPE0000I INITTERM MYCQSIT0          0          1          2
BPE0000I INITTERM OEMCQIT0          0          1        162
BPE0000I OVERFLOW OVERFL01          5          3          6
BPE0000I OVERFLOW OVERFL02          5          3         19
BPE0000I OVERFLOW OVERFL03          5          3          9
BPE0000I OVERFLOW OVERFL04          5          3      15593
BPE0000I STREVENT STREVNT0         10        542        628
BPE0000I STRSTAT  STRSTAT0          1         36       1889
BPE0000I STRSTAT  STRSTA10          1         36        241
BPE0032I DIS USRX COMMAND COMPLETED
```

Command Example 3: Display the entry point, load point, and size of all of the SCI CLNTCONN user exit modules.

Command:

```
F SCI,DIS USRX NAME(CLNTCONN) SHOW(SIZE,ENTRYPT,LOADPT)
```

Output:

```
BPE0030I EXITTYPE MODULE    ENTRYPT   LOADPT     SIZE
BPE0000I CLNTCONN SCCLCN00 8B864D78 8B864D78 00000458
BPE0000I CLNTCONN SCCLCN10 8BA14200 8BA14200 00001C10
BPE0000I CLNTCONN SCCLCN20 8BA18EE8 8BA18AF0 00000AB0
BPE0032I DIS USRX COMMAND COMPLETED
```

Command Example 4: Display the first part of the module text for all of the BPE user exits in the OM address space.

Command:

```
F OM,DIS USRX NAME(*) OWNER(BPE) SHOW(TEXT)
```

Output:

```
BPE0030I EXITTYPE  MODULE   TEXT
BPE0000I INITTERM  MYINIT00 .MYINIT00+20010615+17:47...
BPE0000I STATS     HHGSTAT0 .HHGSTAT0+20010615+08:47...
BPE0032I DIS USRX COMMAND COMPLETED
```

Command Example 5: Display the refresh time for all of the RM INITTERM modules.

Command:

```
F RM,DIS USRX NAME(*) OWNER(RM) SHOW(RTIME)
```

Output:

```
BPE0030I EXITTYPE MODULE   RTIME
BPE0000I INITTERM RMINITRM 2001-06-15 16:48:22.39
BPE0032I DIS USRX COMMAND COMPLETED
```

Refreshing User Exits

The `REFRESH USEREXIT` command causes BPE to reprocess the user exit PROCLIB members specified in the BPE configuration PROCLIB member and to reload the user exit modules currently listed in the user exit PROCLIB members for the types specified on the command. This command enables you to make updates to your user exits without stopping and restarting the address space.

When you enter the `REFRESH USEREXIT` command, BPE performs the following processing:

- Reads any user exit PROCLIB members that are specified on EXITMBR= statements in the BPE configuration proclib member. Because BPE re-reads these members at the time you issue the command, you can edit the user exit PROCLIB members prior to issuing the `REFRESH` command and make changes to the user exit definitions. BPE does *not* re-read the main BPE configuration proclib member, so you cannot change the names of the user exit PROCLIB members, only their contents.
- Loads the user exit modules specified on the EXITDEF= statements for the user exit types specified on the command.
- Quiesces all current user exits. This means that the command waits for any active exits to complete processing and delays any new calls to the current exits. This ensures that no user exit is running while the exit is being refreshed.
- Replaces pointers to the previous user exit modules with pointers to the newly-loaded modules in internal BPE control blocks, which are used to manage the calling of the exits.
- Resumes the user exits and allows calls to be made to the newly-loaded exits.
- Deletes the old copy of the user exits.

BPE loads the new copies of the user exit modules before deleting the old modules. If an error occurs during this process (for instance, a module could not be loaded or BPE internal control block storage could not be obtained), BPE will fail the command and leave the old copies of the user exits in effect. All modules of the specified user exit type(s) must be loaded successfully for the command to complete successfully.

When a user exit module is refreshed, its abend count is reset to zero. This means that a user exit module that had reached its abend limit (specified by the ABLIM parameter on the EXITDEF statement) and was no longer being called by BPE is again called.

Important: If you changed the ABLIM parameter for a user exit in the PROCLIB member, the *new* value of ABLIM takes effect after the refresh command.

Related Reading: See *IMS Version 9: Base Primitive Environment Guide and Reference* for more information about the EXITDEF statement.

Considerations for Refreshing User Exits:

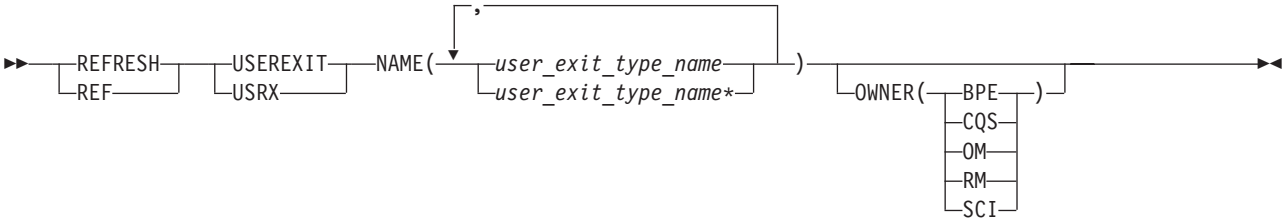
- When you refresh a user exit type, BPE reloads all exit modules defined for that type. The new copies of the modules will be at a different virtual address than the old copies. If your modules are re-entrant, this should not be a problem. However, if your modules are not re-entrant and they store data within themselves, they must be able to tolerate being reloaded and losing the information previously stored within them.

  Recommendation: Code and link edit all user exit modules as re-entrant to avoid this problem.
- If you refresh a user exit module that had previously been loaded, BPE continues to pass the same static work area that the previous copy of the module had been using. If the new version of the module has a different mapping or use of this area than the previous version, the new version must contain toleration code that can handle the old-style formatted data within this static work area.

  Recommendation: Place a version number in the static work area, so that your exits can easily tell when they are using a back-level data structure within this work area.
- If you remove a user exit module from an EXITDEF list and refresh the exits, BPE deletes the static work area associated with that exit module. If you later add the module back to the EXITDEF list and refresh the exits, the module gets a new (cleared) static work area, *not* the work area it had previously.
- If your user exits are being managed by LLA using VLF (or an equivalent product), you must ensure that the copies of the module(s) being refreshed are updated in LLA prior to issuing the `REFRESH USEREXIT` command. See the *MVS Initialization and Tuning Guide* for information on LLA-managed libraries.
- If you have user exits that issue MVS WAITs for long periods of time (for example, a WAIT for an external event that may be delayed, such as a WTOR), then issuing a `REFRESH USEREXIT` command could cause a performance problem or work stoppage. This is because BPE has to quiesce the user exits in order to process the `REFRESH` command. BPE must wait until all currently-called user exits complete before it can perform the user exit refresh. BPE prevents any new calls to user exits until after the command completes. If a user exit has been called and does not return to BPE for a long period of time, the `REFRESH` command is delayed until the exit returns. No other user exits can be called while BPE is waiting, so the processes that are invoking the user exits are also put into a wait.

Recommendation: Ensure that your user exits avoid long WAITs, and avoid issuing services that might WAIT.

Format of REFRESH USEREXIT Command



Usage of REFRESH USEREXIT Command

REFRESH | REF

A required parameter, which specifies that the action against the specified resource(s) is to refresh the resource(s).

USEREXIT | USRX

A required parameter, which specifies that the resource type being acted upon is a BPE-managed user exit type.

NAME(*user_exit_type_name***)**

A required parameter, which specifies the name of the user exit type or types that you want to refresh. You can specify a single user exit type name or a list of user exit type names separated by commas. User exit type names can contain wildcard characters.

Related Reading: For more information about using wildcard characters, see “BPE Wildcard Character Support” on page 816.

Important: The names specified in this parameter are the name(s) of user exit types, *not* the names of individual user exit modules.

BPE and each address space that can use BPE have different user exit types. BPE’s user exit types, as specified by OWNER(BPE), include the following:

INITTERM Initialization-Termination user exit

STATS BPE system functions statistics user exit

User exit types are defined in all CQS address spaces, as specified by OWNER(CQS), and include the following:

CLNTCONN Client Connection user exit

INITTERM Initialization-Termination user exit

OVERFLOW Queue Overflow user exit

STRSTAT Structure statistics user exit

STREVENT Structure event user exit

User exit types are defined in all OM address spaces, as specified by OWNER(OM), and include the following:

CLNTCONN Client Connection command registration and deregistration user exit

INITTERM Initialization-Termination user exit

INPUT Command input user exit

OUTPUT Output user exit

SECURITY Security checking user exit

User exit types are defined in all RM address spaces, as specified by OWNER(RM), and include the following:

CLNTCONN Client Connection and Disconnection user exit

INITTERM Initialization-Termination user exit

User exit types are defined in all SCI address spaces, as specified by OWNER(SCI), and include the following:

CLNTCONN Client Connection and Disconnection user exit

INITTERM Initialization-Termination user exit

Related Reading:

- See *IMS Version 9: Base Primitive Environment Guide and Reference* for more information about the BPE user exit routine types.
- See *IMS Version 9: Common Queue Server Guide and Reference* for more information about the CQS user exit routine types.
- See *IMS Version 9: Common Service Layer Guide and Reference* for more information about the OM, RM, and SCI user exit routine types.

OWNER(BPE | CQS | OM | RM | SCI)

An optional parameter that specifies the owner of the user exit type or types that you want to refresh. You can specify one of the following values:

BPE For all IMS components that are running in a BPE address space.

CQS For CQS address spaces only.

OM For OM address spaces only.

RM For RM address spaces only.

SCI For SCI address spaces only.

The OWNER parameter acts as a filter to help you select the user exit types that you want to refresh. For example, you could specify `NAME(*) OWNER(CQS)` to refresh all of the CQS-defined user exit types in a CQS address space. If `OWNER` is omitted, then both BPE and component user exits can be refreshed (depending on the exits specified on `NAME`).

REFRESH USEREXIT Command Output: The `REFRESH USEREXIT` command output consists of message, BPE0032I indicating that the command has completed:

```
BPE0032I REFRESH USEREXIT COMMAND COMPLETED
```

Command Example 1: Refresh all user exit modules.

Command:

```
F CQS1,REFRESH USEREXIT NAME(*)
```

Output:

```
BPE0032I REFRESH USEREXIT COMMAND COMPLETED
```

Command Example 2: Refresh all user exit modules for the OM command input and output exit types.

Command:

```
F OM,REF USRX NAME(INPUT,OUTPUT)
```

Output:

```
BPE0032I REF USRX COMMAND COMPLETED
```

Part 6. Appendixes

© Copyright IBM Corp. 1974, 2003

Appendix A. DBCTL Commands

Table 191 is a list of commands and keywords valid in the Database Control (DBCTL) environment. All commands and keywords are valid in a DB/DC environment.

Table 191. Commands and Keywords Valid in DBCTL

Commands	Keywords	Page or Topic
/CHANGE	ABORT, AUTOLOGON, CCTL, COMMIT, FDR, OASN, PASSWORD, PRTKN, RESET, SUBSYS, TIMEOUT	123
/CHECKPOINT	ABDUMP, FREEZE, PURGE, STATISTICS	139
/DBDUMP	DATABASE, GLOBAL, LOCAL, NOFEOV, NOPFA	159
/DBRECOVERY	AREA, DATABASE, DATAGROUP, GLOBAL, LOCAL, NOFEOV, NOPFA	163
/DELETE	DATABASE, PASSWORD, PROGRAM	171
DELETE LE	LTERM, PGM, TRAN, USERID	175
/DEQUEUE	AOITOKEN	183
/DISPLAY	ACTIVE, AOITOKEN, AREA, BKERR, CCTL, DATABASE, DBD, FDR, FPVIRTUAL, HSSP, INDOUBT, MADSIOT, MODIFY, MONITOR, OASN, OLDS, OSAMGTF, PI, POOL, PROGRAM, PSB, RECOVERY, REGION, SHUTDOWN, STATUS, SUBSYS, TABLE, TCO, TRACE, TRACKING STATUS, XTRC	195
/ERESTART	CHECKPOINT, COLDBASE, COLDSYS, FORMAT, NOBMP, OVERRIDE	353
INITIATE OLC	OPTION, PHASE, TYPE	385
/LOCK	DATABASE, PROGRAM	381
/LOG		413
/MODIFY	ABORT, COMMIT, PASSWORD, PREPARE	417
/NRESTART	CHECKPOINT, FORMAT	435
/PSTOP	AOITOKEN, JOBNAME, REGION	453
QUERY LE	LTERM, PGM, SHOW, TRAN, USERID	468
QUERY MEMBER	ALL, ATTRIB, SHOW, STATUS, TYPE	468
QUERY OLC	LIBRARY, SHOW	468
/RECOVER	ALLENTRIES, AREA, CAGROUP, DATAGROUP, DB, DBDS, DBDSGRP, ERRORCONT, NOCHECK, OFFLINE, PITR, RCVTIME, RCVTOKEN, READNUM, RECOVGRP, STAGLOBAL, STALOCAL, USEAREA, USEDBDS	545
/RMxxxxxx		565
/RTAKEOVER		579
/SSR		599

© Copyright IBM Corp. 1974, 2003

Table 191. Commands and Keywords Valid in DBCTL (continued)

Commands	Keywords	Page or Topic
/START	ACCESS, AREA, AUTOARCH, DATABASE, DATAGROUP, DBALLOC, GLOBAL, ISOLOG, JOBNAME, LOCAL, MADSIOT, NOBACKOUT, NODBALLOC, OLDS, PROGRAM, REGION, SB, SERVGRP, SSM, SUBSYS, THREAD, TRKAUTOARCH, WADS	601
/STOP	ABDUMP, ADS, AREA, AUTOARCH, CANCEL, DATABASE, DATAGROUP, GLOBAL, JOBNAME, LOCAL, NOPFA, OLDS, PROGRAM, REGION, SB, SERVGRP, SUBSYS, THREAD, WADS	629
/SWITCH	CHECKPOINT, OLDS, WADS	657
TERMINATE OLC		661
/TRACE	COMP, MONITOR, NOCOMP, OPTION, OSAMGTF, PI, PROGRAM, PSB, SET, TABLE, TCO, VOLUME	677
/UNLOCK	DATABASE, PROGRAM	699
UPDATE LE	LTERM, PGM, SET, TRAN, USERID	707
/VUNLOAD		751

Appendix B. DCCTL Commands

Table 192 is a list of commands and keywords valid in a Data Communications Control (DCCTL) environment. All commands and keywords are valid in a DB/DC environment.

Table 192. Commands and Keywords Valid in DCCTL

Commands	Keywords	Page or Topic
/ACTIVATE	LINK, NODE	95
/ALLOCATE	LUNAME, MODE, TPNAME	97
/ASSIGN	CLASS, COMPONENT, CPRI, ICOMPONENT, INPUT, LINE, LMCT, LPRI, LTERM, NODE, NOSAVE, NPRI, OUTPUT, PARLIM, PLMCT, PTERM, REGION, SAVE, SEGNO, SEGSIZE, TRANSACTION, USER, VTAMPOOL	99
/BROADCAST	ACTIVE, LINE, LTERM, MSNAME, NODE, PTERM, SYSID, USER	115
/CANCEL		121
/CHANGE	APPC, ASR, COLDSESS, CPLOG, CCTL, DESCRIPTOR, DIRECTORY, FORCSESS, ID, INTERVAL, LINK, LOGOND, LUNAME, MAXRGN, MODE, NODE, NOSAVE, OASN, PASSWORD, RESET, SAVE, SIDE, SUBSYS, SURVEILLANCE, SYNCLEVEL, SYNCSESS, TIMEOUT, TPNAME, TRANSACTION, TYPE, UOR, USER	123
/CHECKPOINT	ABDUMP, DUMPQ, FREEZE, LEAVEGR, PURGE, QUIESCE, SNAPQ, STATISTICS	139
/CLSDST	FORCE, NODE, USER	145
/COMPT	CNS, CRD, NODE, NOTRDY, PCH, PDS, PRT, RDR, READY, TDS, UDS, USER, VID, WPM1, WPM2, WPM3	149
/CQCHKPT	SHAREDQ, STRUCTURE, SYSTEM	153
/CQQUERY	STATISTICS, STRUCTURE	155
/CQSET	SHAREDQ, SHUTDOWN, STRUCTURE	157
/DELETE	LINE, LTERM, NODE, PASSWORD, PROGRAM, PTERM, TERMINAL, TRANSACTION	171
DELETE LE	LTERM, PGM, TRAN, USERID	175
/DEQUEUE	AOITOKEN, LINE, LTERM, LUNAME, MSNAME, NODE, PTERM, PURGE, PURGE1, SUSPEND, TMEMBER, TPIPE, TPNAME, TRANSACTION, USER	183

© Copyright IBM Corp. 1974, 2003

Table 192. Commands and Keywords Valid in DCCTL (continued)

Commands	Keywords	Page or Topic
/DISPLAY	ACTIVE, AFFINITY, AOITOKEN, APPC, ASSIGNMENT, AUTOLOGON, BALGRP, CLASS, CONVERSATION, CPLOG, CQS, DC, DESCRIPTOR, EMHQ, HSB, INPUT, LINE, LINK, LTERM, LUNAME, MODE, MODIFY, MONITOR, MSGAGE, MSNAME, MSPLINK, NODE, OASN, OLDS, OTMA, OUTPUT, OVERFLOWQ, POOL, PRIORITY, PROGRAM, PSB, PTERM, Q, QCNT, REGION, REMOTE, RTCODE, SHUTDOWN, STATUS, STRUCTURE, SUBSYS, SYSID, TABLE, TCO, TIMEOUT, TIMEOVER, TMEMBER, TPIPE, TPNAME, TRACE, TRACKING STATUS, TRANSACTION, TRAP, UOR, USER, XTRC	195
/END	LINE, NODE, PTERM, USER	351
/ERESTART	BACKUP, BUILDQ, CHECKPOINT, CMDAUTH, CMDAUTHE, COLDCOMM, COLDSYS, FORMAT, MULTSIGN, NOBMP, NOCMDAUTH, NOCMDAUTHE, NOPASSWORD, NOTERMINAL, NOTRANAUTH, NOTRANCMDS, NOUSER, OVERRIDE, PASSWORD, SNGLSIGN, TERMINAL, TRANAUTH, TRANCMDS, USER	353
/EXCLUSIVE	LINE, NODE, PTERM, USER	365
/EXIT	CONVERSATION, LINE, NODE, PTERM, USER	367
/FORMAT	LTERM	371
/HOLD		373
/IAM	DONE, LTERM, PTERM	375
/IDLE	LINE, LINK, NODE, NOSHUT	377
INITIATE OLC	OPTION, PHASE, TYPE	385
/LOCK	LTERM, NODE, PROGRAM, PTERM, TRANSACTION	381
/LOG		413
/LOOPTEST	LINE, PTERM	415
/MODIFY	ABORT, COMMIT, LTERM, PASSWORD, PREPARE, TERMINAL, TRANCMDS	417
/MONITOR	LINE, PTERM	425
/MSASSIGN	LINK, LOCAL, MSNAME, MSPLINK, SYSID, TRANSACTION	427
/MSVERIFY	MSNAME, SYSID	431
/NRESTART	BUILDQ, CHECKPOINT, CMDAUTH, CMDAUTHE, FORMAT, MULTSIGN, NOBUILDQ, NOCMDAUTH, NOCMDAUTHE, NOPASSWORD, NOTERMINAL, NOTRANAUTH, NOTRANCMDS, NOUSER, PASSWORD, SNGLSIGN, TERMINAL, TRANAUTH, TRANCMDS, USER	435
/OPNDST	ID, LOGOND, MODE, NODE, Q, UDATA, USER, USERD	447
/PSTOP	AOITOKEN, CLASS, FORCE, JOBNAME, LINE, LINK, LTERM, MSPLINK, PTERM, PURGE, REGION, TRANSACTION	453

Table 192. Commands and Keywords Valid in DCCTL (continued)

Commands	Keywords	Page or Topic
/PURGE	APPC, CLASS, FPPROG, FPREGION, LINE, LTERM, MSNAME, PTERM, TRANSACTION	461
QUERY LE	LTERM, PGM, SHOW, TRAN, USERID	468
QUERY MEMBER	ALL, ATTRIB, SHOW, STATUS, TYPE	468
QUERY OLC	LIBRARY, SHOW	468
QUERY TRAN	CLASS, NAME, QCNT, SHOW, STATUS	468
/QUIESCE	NODE, USER	537
/RCLSDST		539
/RCOMPT	CNS, NOTRDY, PCH, PDS, PRT, RDR, READY, TDS, UDS, VID	541
/RDISPLAY		543
/RELEASE	CONVERSATION	561
/RESET		563
/RMxxxxxx	LTERM	565
/RSTART	CONTINUOUS, LINE, LINK, LOPEN, MODE, MSPLINK, NODE, PTERM, USER	573
/RTAKEOVER	FREEZE, DUMPQ, NOREVERSE, UNPLAN	579
/SECURE	APPC, OTMA	583
/SET	CONVERSATION, LTERM, TRANSACTION	587
/SIGN		591
/SMCOPY	MASTER, TERMINAL	597
/SSR		599
/START	APPC, AUTOARCH, CLASS, DC, GRSNAME, INPUT, ISOLOG, JOBNAME, LINE, LTERM, LUNAME, MSNAME, NODE, OLDS, OTMA, OUTPUT, PROGRAM, PTERM, REGION, RTCODE, SERVGRP, SSM, SUBSYS, SURVEILLANCE, TMEMBER, TPIPE, TPNAME, TRANSACTION, TRKAUTOARCH, USER, VGRS, WADS	601
/STOP	ABDUMP, APPC, AUTOARCH, BACKUP, CANCEL, CLASS, DC, INPUT, JOBNAME, LINE, LTERM, LUNAME, MSNAME, NODE, OLDS, OTMA, OUTPUT, PROGRAM, PTERM, REGION, RTCODE, SERVGRP, SUBSYS, SURVEILLANCE, TMEMBER, TPIPE, TPNAME, TRANSACTION, USER, VGRS, WADS	629
/SWITCH	ABDUMP, ACTIVE, BACKUP, CHECKPOINT, FORCE, OLDS, SYSTEM, WADS	657
TERMINATE OLC		661
/TEST	LINE, NODE, PTERM, USER	673
/TRACE	AUTO, EXIT, INPUT, LEVEL, LINE, LINK, LUNAME, MODULE, MONITOR, MSG, NODE, OPTION, OUTPUT, PROGRAM, SET, TABLE, TAKEOVER, TCO, TIMEOUT, TMEMBER, TPIPE, TPNAME, TRANSACTION, TRAP, UNITYPE, USER, VOLUME	677

Table 192. Commands and Keywords Valid in DCCTL (continued)

Commands	Keywords	Page or Topic
/UNLOCK	LTERM, NODE, PROGRAM, PTERM, SYSTEM, TRANSACTION	699
UPDATE LE	LTERM, PGM, SET, TRAN, USERID	707
UPDATE TRAN	CLASS, CPRI, LCT, LPRI, MAXRGN, NAME, NPRI, PARLIM, PLCT, SCOPE, SEGNO, SEGSZ, SET, START, STOP	707

Appendix C. List of Reserved Words

Table 193 is a list of words that cannot be used to name resources such as transactions or databases. The words listed below are used and reserved only for IMS commands.

Table 193. Words Used and Reserved Only for IMS Commands

A	ABDUMP	ABORT	ACCESS
ACT	ACTIV	ACTIVATE	ACTIVE
ADDS	ADS	AFF	AFFIN
AFFINITY	AFTER	AOITKN	APDB
APMQ	AOITOKEN	ALL	ALLENTRIES
ALLENT	ALLOC	ALLOCATE	ALLOCF
ALLOCS	APPC	ARCHIVE	AREA
ASMT	ASR	ASSIGN	ASSIGNMENT
AUTO	AUTOARC	AUTOARCH	AUTOLOGON
AUTOLGN	AUTOSR	BACKOUT	BACKUP
BALG	BALGRP	BKERR	BLDQ
BLDQS	BROADCAST	BU	BUILDQ
BUILDQS	C1INOP	C2INOP	C3INOP
C4INOP	CAGROUP	CAGRP	CANCEL
CC	CCTL	CHANGE	CHECKPOINT
CHECKPT	CHKPOINT	CHKPT	CHNGS
CLASS	CLS	CLSDST	CMDAUTH
CMDAUTHE	CNS	COLDB	COLDBASE
COLDC	COLDCOMM	COLDS	COLDSESS
COLDSYS	COMMIT	COMP	COMPINOP
COMPONENT	COMPT	CON	CONT
CONTINUOUS	CONV	CONVACT	CONVERSATION
CONVHLD	CPRI	CQCHKPT	CQQUERY
CPLOG	CQC	CQQ	CQS
CQSET	CRD	DATABASE	DATABASES
DATAGROUP	DATAGRP	DB	DBALLOC
DBD	DBDS	DBDSGRP	DBDUMP
DBR	DBRECOVERY	DBS	DC
DEACT	DEADQ	DELETE	DELS
DEQUEUE	DESC	DESCRIPTOR	DL/I
DLOG	DIS	DISP	DISPLAY
DIR	DIRECTORY	DMS	DONE
DUMPQ	DUMPQS	EEQE	EMHQ
END	ERESTART	ERRORCONT	EXCL
EXCLUSIVE	EXIT	EXTRACE	FAST
FDR	FIRST	FMS	FMT
FOR	FORC	FORCE	FORCES
FORCSESS	FORMAT	FPPROG	FPREGION
FPRGN	FPV	FPVIRTUAL	FREEZE
GLOBAL	GRS	GRSN	GRSNAME
HOLD	HOTSTANDBY	HSB	HSSP
IAM	IC	ICOMPONENT	ICOMPT
ID	IDCO	IDLE	IND
INDOUBT	INOP	INPUT	INQONLY
INT	INTERVAL	INTV	IOVF
ISOLOG	JBN	JOB	JOBNAME
KEY	LA	LATC	LCT
LEAVEGR	LEVEL	LGND	LINE

© Copyright IBM Corp. 1974, 2003

Table 193. Words Used and Reserved Only for IMS Commands (continued)

LINES	LINK	LMCT	LOC
LOCAL	LOCK	LOG	LOGOND
LOOPTEST	LOPEN	LOST	LPRI
LRTT	LTERM	LTERMS	LU
LUMI	LUN	LUNAME	MADSIOT
MASTER	MAXRGN	MESSAGE	MFST
MFSTEST	MODE	MODETABLE	MODETBL
MODIFY	MODS	MODULE	MON
MONITOR	MSASSIGN	MSDB	MSDBLOAD
MSG	MSGAGE	MSGREG	MSGREGION
MSGREGIONS	MSGREGS	MSNAME	MSPLINK
MSVERIFY	MULTSIGN	NBLDQ	NOBACKOUT
NOBKO	NOBLDQ	NOBMP	NOBUILDQ
NOCHECK	NOCMDAUTH	NOCMDAUTHE	NOCOMP
NODBALLOC	NODE	NOFEOV	NOIN
NONE	NOOUT	NOPASSWORD	NOPFA
NOPSWD	NOQUEUE	NOREVERSE	NOS
NOSAVE	NOSHUT	NOTER	NOTERM
NOTERMINAL	NOTINIT	NOTOPEN	NOTRANAUTH
NOTRANCMDS	NOTRDY	NOUSER	NPRI
NRESTART	OASN	OFF	OFFLINE
3 OFR	OLDS	OLREORG	ON
OPNDST	OPTION	OSAMGTF	OTMA
OTMT	OUTPUT	OVER	OVERFLOWQ
OVERRIDE	OVFLWQ	PAGE	PARLIM
PASSWORD	PASSWORDS	PCH	PDS
PGM	PGMS	PI	PITR
PLCT	PLMCT	POOL	PREL
PREO	PREPARE	PRI	PRIMARY
PRIORITY	PRK	PROG	PROGRAM
PROGRAMS	PROGS	PRST	PRT
PRTKN	PRTY	PSB	PSS
PSTOP	PSTOPPED	PSWD	PSWDS
PTERM	PTERMS	PUR	PURGE
PURGE1	Q	QCNT	QMGR
QRTT	QS	QUEUE	QUEUES
QUI	QUIESCE	RCLSDST	RCOMPT
RCS	RCVTIME	RCVTOKEN	RDISPLAY
RDR	READNUM	READY	RECOVERY
RECOVGRP	REG	REGION	REGIONS
REGS	RELEASE	RELREQ	REMOTE
REMOVE	RESET	RESP	RESPINP
RESYNC	RETR	RMCHANGE	RMDELETE
RMGENJCL	RMINIT	RMLIST	RMNOTIFY
RSTART	RTAKEOVER	RTC	RTCODE
SAVE	SB	SCHD	SEC
SECURE	SECURITY	SEGNO	SEGSIZE
SEGSZ	SERVGRP	SET	SG
SHAREDQ	SHRQ	SHUT	SHUTDOWN
SIDE	SIGN	SIMLOGON	SINGLE
SMCOPY	SNAPQ	SNGL	SNGLSIGN
SQTT	SSM	SSR	STAGLOBAL
STALOCAL	STA	START	STATIC
STATISTICS	STATUS	STO	STOP
STOPPED	STRG	STRUC	STRUCTURE

Table 193. Words Used and Reserved Only for IMS Commands (continued)

SUB	SUBS	SUBPOOL	SUBSYS
SUBSYSMEMBER	SUBSYSTEM	SUBSYSTEMS	SUR
SURV	SURVEIL	SURVEILLANCE	SUSPEND
SWITCH	SYNC	SYNCLEVEL	SYNCLV
SYNCSESS	SYSID	SYSTEM	TABLE
TAKEOVER	TCO	TDS	TER
TERM	TERMINAL	TERMINALS	TERMINATE
TERMS	TERS	TEST	THREAD
TIMEOUT	TIMEOVER	TIMO	TKO
TKOTRA	TKOVR	TMEM	TMEMBER
TO	TP	TPI	TPIPE
TPN	TPNAME	TRA	TRACE
TRACKING	TRACKING STATUS	TRAN	TRANAUTH
TRANCMDS	TRANCODE	TRANCODES	TRANS
TRANSACTION	TRANSACTIONS	TRAP	TRKARCH
TRKAUTOARCH	TRS	TYPE	UDATA
UDS	UNITYPE	UNL	UNLOCK
UNPLAN	UOR	USEDBDS	USER
USERD	USRD	VERIFY	VGR
VGRS	VID	VIR	VOLUME
VPL	VTAMPOOL	VUNLOAD	WADS
WPM1	WPM2	WPM3	XKEY
XTRC	XTRACE		

Appendix D. Shared Secondary Index Database Commands

The original advantage of a shared index database was that it saved a significant amount of main storage for buffers and some control blocks. However, when VSAM was enhanced with shared resources, the savings in storage became less significant. Now the possible disadvantages of using shared index databases generally outweigh the small amount of space they save.

Commands sometimes operate differently depending on whether they are issued for the first of the secondary indexes or for subsequent secondary indexes. The first secondary index is the first database name specified in the DBDUMP statement of the shared secondary index DBDGEN. This first database is the **real** database. Other secondary index databases are physically part of the **real** database but they are logically distinct.

The first column in the table below lists the issuing command, the second column lists where the command is issued, the third column lists the affects of the command that was issued, and the fourth column provides additional comments.

Issuing the Commands...	On the...	Affects...	Comments
/STOP /LOCK	First secondary index	Only the named database	If no applications are scheduled on any shared secondary indexes that cause the authorization of the **real** database by DBRC, the commands have the same effect as the /DBRECOVERY command on the first secondary index. When a /DISPLAY command is issued on the shared secondary index database, the subsequent secondary indexes are shown as stopped or locked only if the /STOP, /LOCK, or /DBRECOVERY command was issued. To undo the /STOP or /LOCK command, issue a /START or /UNLOCK command on the first secondary index.
/STOP /LOCK	Subsequent secondary indexes	Only the named database	To undo the /STOP or /LOCK command, issue a /START or /UNLOCK command on the named database.
/DBDUMP	First secondary index	All databases sharing the secondary index data set	The /DBDUMP command quiesces activity on all the indexes in the shared database. The database is then closed and reopened for input only. To undo the /DBDUMP command, issue a /START command on the first secondary index.
/DBDUMP	Subsequent secondary indexes	Only the named database	The secondary index is available for read only. To undo the /DBDUMP command, issue a /START command on the named database.

© Copyright IBM Corp. 1974, 2003

Issuing the Commands...	On the...	Affects...	Comments
/DBRECOVERY	First secondary index	All databases sharing the secondary index data set	The /DBRECOVERY command quiesces activity on all the indexes in the shared database. The database is then closed and stopped. When the DISPLAY command is issued on the shared secondary index database, the subsequent secondary indexes are shown as stopped or locked only if the /STOP, /LOCK, or /DBRECOVERY command was issued. To undo the /DBRECOVERY command, issue a /START command on the first secondary index.
/DBRECOVERY	Subsequent secondary indexes	Only the named database	This command is the same as the /STOP command for the named database. However, the /DBRECOVERY command works immediately, but the /STOP command allows current work to quiesce. To undo the /DBRECOVERY command, issue a /START command on the named database.

Appendix E. Commands with the NODE USER Keyword Combination

Many commands using the NODE USER keyword combination can be used with dynamic terminals as well as with non-ISC nodes. Using the NODE USER combination from a dynamic terminal ensures that the command is valid only if the USER is still signed on to the NODE. Commands with the NODE USER keyword pair are valid only if:

- The USER is signed on to the NODE
- In an ISC environment, the USER is allocated to the NODE

Commands with the NODE USER keyword pair are only valid for existing nodes and users. No temporary control blocks are created to retain status data.

The following commands are valid for ISC and non-ISC nodes and users:

- `/CHANGE NODE p1 USER p2,...pn/all (ASR ON/OFF)/MODETABLE(modname)`
- `/CLSDST NODE p1 USER p2...pn/all`
- `/COMPT (1/2/3/4) NODE p1 USER p2 (READY/NOTRDY)`
- `/DEQUEUE NODE p1 USER p2 (LTERM p3)`
- `/DISPLAY ASMT NODE p1 USER p2 (LTERM p3)`
- `/DISPLAY NODE p1 USER p2,...pn/all`
- `/END NODE p1 USER p2`
- `/OPNDST NODE p1 USER p2...`
- `/STOP NODE p1 USER p2,...pn/all`
- `/TRACE NODE p1 USER p2,...pn/all`

The following commands are valid for ISC nodes only:

- `/CHANGE NODE p1 USER p2,...pn/all (FORCSESS/SYNCSESS)`
- `/QUIESCE NODE p1 USER p2,...pn/all`

The following commands are valid for ISC, LUP, and 3600 nodes only:

- `/DEQUEUE NODE p1 USER p2 PURGE/LTERM p3 PURGE1`
- `/EXIT (CONV) NODE p2 USER p3`
- `/RSTART NODE p1 USER p2,...pn/all`
- `/START NODE p1 USER p2,...pn/all`
- `/TEST (MFS) NODE p1 USER p2,...pn/all`

© Copyright IBM Corp. 1974, 2003

Appendix F. Commands That Are Valid in ETO

Table 194 is a list of the commands that are valid for lterms, nodes, and users in ETO.

Table 194. Commands Valid in ETO

Commands	Dynamic Lterms That Are Valid	Dynamic Nodes That Are Valid	Dynamic Users That Are Valid
/ACTIVATE		X	
/ASSIGN	X[1]		X[2]
/BROADCAST	X	X	X
/CHANGE		X	X
/CLSDST		X	X
/COMPT		X	X
/DEQUEUE	X	X	X
/DISPLAY	X	X	X
/END			X
/EXCLUSIVE			X
/EXIT		X	X
/FORMAT	X		
/IDLE		X	
/LOCK	X	X	
/MODIFY	X		
/OPNDST		X	
/PSTOP	X		
/PURGE	X		
/QUIESCE		X	X
/RMxxxxxx	X		
/RSTART		X	X
/SET	X		
/SIGN			X
/START	X	X	X
/STOP	X	X	X
/TEST			X
/TRACE		X	
/UNLOCK	X	X	

Notes:

1. Only valid for /ASSIGN LTERM USER.
2. Only valid for /ASSIGN USER.

© Copyright IBM Corp. 1974, 2003

Appendix G. Status and Attributes for the /DISPLAY Command

A list of the attributes and status shown in the /DISPLAY command are shown in Table 195

Table 195. Attributes and Status in the /DISPLAY Command

Status or Attribute	Meaning
	Indicates no status for the resource.
ACCEPT TRAFFIC	Indicates the OTMA client is ready.
ACT MODETBL	Indicates mode table name actually used to initiate mode.
ACTIV	Indicates this node is in an XRF session, on the active system.
ACTIVE, A	Indicates conversation in progress. For /DISPLAY MODIFY, indicates that the routing code (RTCODE) named is active or that the library is active. Indicates the IMS subsystem is the active subsystem. Indicates the current XRF surveillance mechanism is active. Indicates the OTMA client is in an active XCF group. Indicates the secondary master terminal is active.
ACTIVE-DBCMD	Indicates an /DBD or /DBR command is in progress and waiting for the region to terminate before the /DBD or /DBR can complete.
ACTV	Indicates, for a logical link, that link startup processing is complete and the line is available for message transfer.
ALLOC	Indicates that an ISC user is allocated. The node name to which the user is allocated or signed on follows in parenthesis.
ALLOCATION IN PROGRESS	Indicates a conversation is in the process of being allocated.
ALLOCF	Indicates that database allocation failed.
ALLOCS	Indicates that database allocation was successful.
AREA	The subset of a DEDB.
AUTOSR	Indicates a session initiation option of ASR.
AVAILABLE	Indicates the OLDS can be reused. Indicates a region is available to schedule an application. Indicates a coupling facility structure is available for use by IMS.
AWAITING RESPONSE	Indicates an active terminal is waiting for a response.
AWAITING SNAPQ	Indicates the alternate subsystem is waiting for /CHE SNAPQ from the active subsystem.
BACKOUT	Indicates the OLDS is potentially required for backout or, in the case of a database, there are incomplete backouts preventing the use of the database.
BACKUP	Indicates he IMS subsystem is the alternate subsystem.
BAL	Identifies a transaction eligible for load balancing (for example, with parallel limits specified). The current maximum number of regions that can be simultaneously scheduled to process the transaction follows in parentheses.
BCKUP	Indicates this node is in an XRF session, on the alternate system.

© Copyright IBM Corp. 1974, 2003

Table 195. Attributes and Status in the /DISPLAY Command (continued)

Status or Attribute	Meaning
C1INOP, C2INOP, C3INOP, C4INOP	Indicates the inoperable node or terminal component, where C1, C2, C3, and C4 refer to different components as defined by system definition. (Refer to the /COMPT and /RCOMPT commands for details on how to ready inoperable components.)
CANCEL	Indicates an APPC connection was stopped by the /STOP APPC CANCEL command.
CHECK	Indicates the OTMA client is using RACF for security verification.
CLSER	Indicates an error has occurred when closing the OLDS.
CLSDST	Indicates that a /CLSDST or /STOP command has been entered for a VTAM node but has not yet taken effect. This condition can occur because of an outstanding input or output operation that has not completed. The VTAM network operator can issue an z/OS VARY command to clear the condition.
COLD	Indicates, for a logical link, that link startup processing is not complete.
COMPINOP	Indicates an inoperable terminal component. For 3270 terminals, this is an indication the either the display or the printer is not ready.
CON	Indicates that a node is connected, or in session, with IMS.
CONN	Subsystem connection.
CONN, ACTIVE	Indicates an application program has established communication with a subsystem.
CONNECTED	Indicates IMS is connected to a coupling facility structure.
CONN IN PROGRESS	Subsystem connection in progress.
CONVACT, CONV-ACT	Indicates an active conversation on this terminal, node, or user.
CONVERSATION TERM/USER	Indicates for /DISPLAY MODIFY that the transaction to be changed or deleted or the transaction referencing a program or database to be changed or deleted is in conversation. The terminal originating the conversational transaction and the conversation ID are also displayed. The terminal displayed is: • **nodename** for VTAM terminals • **nodename** and **username** if an ETO user is signed on to the node • **username** for ETO users who signed off while in conversation • **lin#-pte#** (line and pterm number) for BTAM terminals • **1- SC** for the system console
CONVHLD, CONV-HLD	Indicates a held conversation on this terminal, node, or user.
COPY-PHASE	Indicates the CREATE utility is currently in COPY-PHASE for an ADS.
CREATED	Indicates the XCF group has been created, but the OTMA client has not yet connected to it.
DB-STOPD	Indicates a database used by this program is stopped.

Table 195. Attributes and Status in the /DISPLAY Command (continued)

Status or Attribute	Meaning
/DBD ACTIVE	A /DBDUMP command is in progress for a database to be changed or deleted.
/DBR ACTIVE	A /DBRECOVERY command is in progress for a database to be changed or deleted.
DEACT (DEACTIVATED)	Indicates a physical terminal/node or line has been permanently deactivated. Restart of BTAM node requires /STOP DC and /START DC commands. Message DFS2473 in the system console log might contain information regarding the reason this status was set. DFS2473 can occur more than once in the system console log.
DEADQ	Indicates a user with dead letter queues or whose last access time was outside the limit set by the DLQT JCL parameter. The DEADQ status can be removed by signing on the user or entering the /DEQUEUE or /ASSIGN command. For a further description and methods to reset the DEADQ status, refer to *IMS Version 9: Administration Guide: Transaction Manager*.
DEALLOCATION IN PROGRESS	Indicates a conversation is in the process of being deallocated.
DEQCT	Number of local message dequeues since the last IMS cold start or, in a shared-queue environment, since the last IMS restart. DEQCT is reset to zero when ENQCT reaches 32768. A DEQCT count can equal the sum of the DEQCTs of multiple destinations (logical terminals or transactions).
DISABLED	Indicates APPC/IMS is not known to APPC/z/OS.
DISCONNECTED	Indicates a node is not in session. Indicates the OTMA client has disconnected from the XCF group.Indicates IMS is not connected to a coupling facility structure.
DQF	Indicates a dequeue request for OTMA REPresynch command failed.
DYNAMIC	A transaction was built in a shared-queues environment to allow transaction messages to be enqueued, but the transaction cannot run on the current IMS subsystem.
EEQE	Extended error queue element, indicates that one or more error queue elements are associated with the database.
ENABLED	Indicates APPC is started.
ENQCT	Number of local message enqueues since the last IMS cold start, or in a shared-queue environment, since the last IMS restart. ENQCT is reset to the number of messages on queue (QCT) when the counter reaches 32768 (QCT = ENQCT − DEQCT). The ENQCT can exceed 32768 if the DEQCT remains zero. The ENQCT will be reset to zero if the counter exceeds 65535. An ENQCT count can equal the sum of the ENQCTs of multiple destinations (logical terminals or transactions).
ERE	Indicates, for a logical link, that link startup processing is not complete. When the link is started, emergency restart synchronization will be performed, because the previous link shutdown was either not normal or an IMS emergency restart was performed.
EXCL	Indicates a node, terminal, or user is in exclusive mode.

Table 195. Attributes and Status in the /DISPLAY Command (continued)

Status or Attribute	Meaning
FAILED	Indicates APPC failed to start. Indicates OTMA failed to start or that the XCF group failed initialization.
FILLING	Indicates a log gap is currently being filled.
FORCE, FORCES	Indicates a session initiation option of FORCE.
FORMAT-PHASE	Indicates the CREATE utility is currently in FORMAT-PHASE for an ADS.
FULL	Indicates the OTMA client is using RACF for security verification, including for dependent regions.
GLOBAL QUEUE COUNT INTERNAL ERROR, GLOBAL QUEUE COUNT STORAGE ERROR	Indicates IMS is unable to access global queue counts or obtain storage to query the queue counts.
IC	Indicates image copy is in progress for that area or database by an HSSP region.
IDENTIFICATION IN PROGRESS	Indicates that an the IMS logger is in the process of identifying to the Transport Manager Subsystem but has not yet completed.
IDENTIFIED	Indicates that an IMS logger has identified to the Transport Manager Subsystem.
IDLE	Indicates that no activity of any kind is in progress for a line, node, or logical link. This is a common condition for VTAM node channel-to-channel links, and processor storage-to-processor storage links.
INACTIVE, I	Indicates the current XRF surveillance mechanism is inactive. For /DISPLAY MODIFY, indicates the library is inactive. The RSR tracking subsystem is idle.
INOP	Indicates a terminal or node is inoperable.
IN-OVERFLOW	Indicates a coupling facility structure is in overflow mode.
INPUT IN PROGRESS	Indicates input in progress for an active terminal.
INQONLY	Indicates a /DBDUMP command was issued for the database.
IN SLOWDOWN	Indicates the OTMA client is experiencing a slowdown due to excessive message traffic or resource constraints.
INUSE, IN USE	Indicates that queuing is in process for this transaction. (either terminal input or program-to-program switch). Also indicates this is the OLDS currently being used.
INVALID SUBSYSTEM NAME = XXXX	Indicates that a subsystem name was not defined to IMS.
I/O PREVEN, I/O PREVENT, I/O PREV	Indicates a BMP program containing GSAM cannot complete scheduling because I/O prevention has not completed. Indicates further I/O requests to data sets are inhibited.
I/O TOLERATION	Takeover process by which an alternate IMS subsystem ensures database integrity and enables new transaction processing as soon as possible.
LOCK	Indicates a node, terminal, transaction, program, or database is locked.
LOOPTEST	Indicates a line or terminal in looptest mode.

Table 195. Attributes and Status in the /DISPLAY Command (continued)

Status or Attribute	Meaning
LOST	Indicates the VTAM LOSTERM EXIT has been scheduled for this node but has not yet been recognized by IMS. At the next interrupt for this node, IMS will interrogate the LOSTERM value. All values, with one exception, result in an immediate CLSDST, or disconnection, from IMS. For the LOSTERM exception, IMS must wait for VTAM to notify IMS (by another LOSTERM) of completion of recovery operation. Indicates an IMS logger's connection to the Transport Manager Subsystem is gone due to TMS or VTAM failure.
MESSAGES WAITING	Indicates there are system messages waiting to be sent, which prevents shutdown from completing.
MFSTEST, MFST	Indicates a terminal, node, or user in MFSTEST mode.
MSGS IN QUEUE	Indicates there are messages in the queue for an active terminal.
MSG CT	Number of messages on the queue for this destination (calculated by subtracting the DEQCT from ENQCT).
N/A	Indicates a link is not assigned or a user is signed on to a static terminal, in which case queues are not applicable.
NEEDED	Indicates the OLDS that needs to be archived.
NO ACTIVE CONVERSATION	Indicates there is no active conversation between a logger and a log router.
NODE-USR	Node name and user identifier.
NOIN	Indicates a line or terminal stopped for input.
NO INPUTTING LINES	Indicates no terminal activity.
NO LOG ROUTER SG DEFINED	Indicates that there is no Log Router Service Group defined to DBRC.
NONE	Indicates the OTMA client is using not using RACF for security verification.
NOOUT	Indicates a line or terminal stopped for output.
NO OUTPUTTING LINES	Indicates no terminal activity.
NOQUEUE	Indicates a line or terminal that has had message queuing stopped for message switching.
NOT CONN	No subsystem connection.
NOT DEFINED	Indicates the OTMA client has left the XCF group.
NOT IDENTIFIED	Indicates an IMS logger is not identified to the Transport Manager Subsystem.

Table 195. Attributes and Status in the /DISPLAY Command (continued)

Status or Attribute	Meaning
NOTIDLE s	Indicates that a logical link is waiting for the completion of a synchronous event. The s represents the following subcodes: **A** Indicates a status of NOTIDLE-POST, which means that an event has completed but the link has not been dispatched to process it. **B** Indicates a status of NOTIDLE-IWAIT, which means the link is waiting for completion of internal I/O. **Cxx** Indicates a status of NOTIDLE-TP WAIT, which means the link is waiting for completion of a TP access method request. The two characters xx indicate the value of the access method operation code. Blanks appear if the link is VTAM.
NOTINIT	Indicates a database or program for which directory initialization failed or indicates DMB or PSB pool space not large enough to accommodate blocks.
NOTOPEN, NOT-OPEN	Indicates a database, area, line, terminal, or physical link that is not in open status. This status is not applicable to MSDBs.
NOT USABLE	Indicates a log data set (OLDS) is unusable because of previous error.
NRE	Indicates, for a logical link, that link startup processing is not complete. When the link is started, normal restart synchronization will be performed, because the previous link shutdown or IMS restart was normal.
OFR	Indicates that the database or area is being brought up to the current tracking level with online forward recovery.
OPEN	Indicates that the VTAM ACB is open. Indicates an area to be changed or deleted is open.
OPNDST	Indicates an OPNDST is in process for this node.
OUTBOUND	Indicates the alternate subsystem is APPC-enabled. Also shown on active subsystem when APPC processing cannot start.
OUTPUT IN PROGRESS	Indicates output in progress for an active terminal.
PAGE	Indicates an MFS paged message.
PERMNT ERR	Indicates there is a permanent error, such as a read error, that prevents a log gap from being filled.
PLANNED TAKEOVER IN PROGRESS	Indicates all active subsystems being tracked have successfully shut down and the tracker is now in the process of shutting down.
PRE-OPEN FAILED	Pre-open for ADS failed.
PREV CLSER	Indicates that the previous OLDS could not be closed because of an I/O error; therefore, this OLDS is required by the Log Recovery utility to correct the condition.
PRI	Indicates this node is the primary partner of an ISC session; or, for a link, indicates it is the primary partner of an MSC-VTAM session.
PRIMARY MSTR	Indicates the terminal is the primary master terminal.

Table 195. Attributes and Status in the /DISPLAY Command (continued)

Status or Attribute	Meaning
PROFILE	Indicates the OTMA client is using values in the Security Data section of the message prefix for a transaction.
PRST	Indicates that a terminal or user is in preset destination mode. The destination trancode or logical terminal name follows in parentheses.
PSB SCHEDULED	A program referencing a database to be changed or deleted is scheduled.
PSTOPPED (PSTO, PSTOP)	Indicates a line, terminal, transaction, or logical link that has been pstopped. This status can indicate that a different, necessary resource is unavailable. For example, a transaction could show a status of PSTOP because the associated program is not initialized.
PUR, PURGING	Indicates a transaction, line, or terminal that is purging all of its output.
QCT	Number of messages on the queue for this destination (calculated by subtracting the DEQCT from ENQCT).
QERR, QERROR	Indicates that an I/O error has occurred on the queue for this LTERM, or remote transaction (MSC).
QLOCK	Indicates that the LTERM is locked from sending any further output or from receiving input which could create additional output for the same LTERM until the state is reset by a specific request received on the session.
QUEUING nn	Indicates that messages are queued to the transaction to be changed or deleted, and nn is the number of messages queued.
QUI	Indicates that a VTAM node has sent a VTAM Quiesce-End-of-Chain indicator to suspend IMS output.
QUIESCED	Indicates the XCF group is stopped. Indicates the node is stopped.
REBLD-INPROG	Indicates a CQS structure rebuild is in progress for a structure.
RECALL	Indicates database or area is in recall.
RECOVERY	Requests the display of recovery values that pertain to the node or user. For the `/DISPLAY DATABASE`command, indicates that ORS recovery is in progress for that database.
RECOVERY-NEEDED, RECOVERN	Indicates areas that need recovery.
RELREQ (NODE)	Indicates the VTAM RELREQ exit routine has been driven but IMS is waiting for an operation in progress to complete before releasing the node.
REP	Indicates IMS is waiting for an OTMA REPresynch command from the client.
REQ	Indicates IMS is sending an OTMA REQresynch command to the client.
RESP	Indicates the node, line, terminal, or user is in response mode and the response reply message is available for output or in the process of being sent.

Table 195. Attributes and Status in the /DISPLAY Command (continued)

Status or Attribute	Meaning
RESPINP, RESP-INP	Indicates the terminal, line, node, or user is in response mode and the response mode input is still in-doubt; for example, the response reply message is not available for output.
RESYNC	Indicates that the positive acknowledgement for an IMS recoverable output message was not received when the connection with the VTAM node was terminated. This message will be subject to resynchronization when the next connection for this node is attempted.
RNL	Randomizer not loaded for a DEDB database.
RSF	Indicates a reset request failed for an OTMA REPresynch command.
SCHEDULED	Indicates that a conversation, transaction, or program has been received and queued. For a conversation, this status will be displayed from the time an input message is entered until the output message is dequeued. With /DISPLAY MODIFY, indicates that the named resource (a transaction or program to be changed or deleted, or a program referencing a database to be changed or deleted) is scheduled. With /DISPLAY OLDS, indicates an archive job had been generated.
SEC	Indicates this node is the secondary partner of an ISC session; or, for a link, indicates it is the secondary partner of an MSC-VTAM session.
SECOND MSTR	Indicates the terminal is the secondary master terminal.
SENDING LOGS	Indicates a logger has an active conversation with a log router and is sending logs to the tracking site.
SERVER	Indicates the OTMA client is acting as the server.
SEVERE-ERROR	Indicates ADS encountered a severe error.
SHUT	Indicates that normal processing has completed for the node and a VTAM shutdown-complete indicator was returned to IMS. The node can receive IMS output but cannot enter data while in this state.
SHUTDOWN-STRCHKPT	Indicates CQS will take a structure checkpoint during normal shutdown.
SIGN	Indicates that a terminal or user is signed on to a node under enhanced security. The user signed on to the node is shown in parentheses following SIGN.
SIMLOGON	Indicates that a logon to IMS has been simulated.
SPND	Indicates that a transaction had messages on the suspend queue.
/STA ACTIVE	A /START DATABASE command is in progress for a database to be changed or deleted.
STARTING	Shown after /DIS CONV is issued and before the status is set to SCHEDULED. Indicates that a conversation has been received but is not eligible for scheduling to an application program until an end-of-message indication is received. Also shown after /START APPC is issued and before the status is set to ENABLED or FAILED.

Table 195. Attributes and Status in the /DISPLAY Command (continued)

Status or Attribute	Meaning
STATIC	Indicates that the node, LTERM or user was defined during system definition.
STOP IN PROGRESS	Indicates a /STOP SUBSYS command is in progress.
STOPPED, STO, STOP	Indicates an area, line, LTERM, LU name, node, terminal, user, OLDS, subsystem, transaction, routing code (RTCODE), OTMA client, program, or database that is stopped. Indicates a subsystem connection is stopped. Indicates a /STOP SERVGRP command was issued.Indicates a transaction pipe is stopped.
STO-INP	Indicates the terminal is stopped with input messages enqueued to the LU name.
STO-OUTP	Indicates the terminal is stopped with output message enqueued to the LU name.
STRCHKPT-INPROG	Indicates a CQS structure checkpoint is in progress for a structure.
SUBSYSTEM XXXX NOT DEFINED BUT RECOVERY OUTSTANDING	Indicates that the subsystem was not defined but IMS has outstanding recovery elements.
SUSPENDED	Indicates that the transaction has been suspended. For /DISPLAY MODIFY, indicates that the transaction to be changed or deleted is on the suspend queue.
SUSPENDED LOGS	Indicates a logger has an active conversation with a log router but has suspended sending logs to the tracking site due to resource shortage. The conversation is still intact.
SYNCHRONIZING, SYN	Indicates the alternate subsystem is processing a /CHE SNAPQ command from the active subsystem. Indicates a transaction pipe is being synchronized.
SYS CONSOLE	Indicates the terminal is the system console.
TAKEOVER IN PROGRESS	Indicates the alternate subsystem is taking over workload from the active subsystem.
TAKEOVER REQUESTED	Indicates the active subsystem has requested a takeover by the alternate subsystem.
TBR	Indicates IMS is waiting for an OTMA TBResynch command from the client.
TERM IN PROGRESS	Indicates an internal termination of the subsystem.
TERMINATING	Indicates the application program is being terminated.
TEST	Indicates a line, node, terminal, or user in test mode.
TKOTRA	Indicates that a node, line, link, or terminal in an XRF session is to be traced only during takeover, to help diagnose XRF terminal switch problems.
TMP	Indicates a transaction pipe is temporary.
TRA	Indicates the physical terminal, node, logical link, LU name, transaction, program or transaction pipe is being traced.
TRACKING	Indicates the XRF alternate subsystem is tracking the active subsystem or an RSR tracking subsystem is tracking one or more active subsystems.

Table 195. Attributes and Status in the /DISPLAY Command (continued)

Status or Attribute	Meaning
UNAVAILABLE, UNAVAIL	Indicates an ADS that is unavailable. Indicates a region is unavailable because an application is using it, even though the application is not currently scheduled. The region is not available to any other application for use. Indicates that there are no records of a log gap at the active site; these records may have been deleted. Indicates a coupling facility structure is not available for use by IMS.
UNKNOWN	Indicates the XCF group has an unknown status.
UNPLANNED TAKEOVER IN PROGRESS	Indicates an /RTAKEOVER command was entered on the tracking subsystem and an unplanned takeover is in progress.
USTOP, USTOPPED	Indicates scheduling of transactions has been stopped due to unavailable data.
VIR	Indicates that the DEDB area has the VSO option.

Table 195. Attributes and Status in the /DISPLAY Command (continued)

Status or Attribute	Meaning
WAIT cond	Indicates the application program is waiting for the completion of an event. The cond represents the reason for waiting. The cond can be: **AOI** indicates that an AO application issued a GMSG call with the wait option specified, but there are no messages for the application to retrieve. **BLOCKMOVER or BLKMVR** indicates that an application control block cannot be loaded because the ACB block mover is busy. **CMD/PENDING** indicates that a /DBD command or a /DBR command is in progress. **INPUT** indicates that the application program is in wait-for-input (WFI) mode. **INTENT or INTENT/POOL** indicates one of two conditions: 1. The application program's intent for a database conflicts with the use of the database by a scheduled program. 2. A temporary shortage of DMB, PSB, of PSB work area pool space exists. **I/O PREVEN** indicates that a BMP region which accesses a GSAM database cannot schedule until I/O prevention has completed. **MESSAGE** indicates that the application program is in a pseudo wait-for-input mode. The application is scheduled and is waiting for a message. **POOLSPACE or POOLSP** indicates a temporary shortage of DMB, PSB, or PSB work area pool space exists. **SWITCHOVER or SWITCH** indicates that the alternate system is tracking the active system. **SYNCPOINT** indicates that the application in the region is now in sync point. **WAIT-EPCB POOL** indicates that there is a temporary shortage of EPCB pool space. **WAIT-RRS/OTMA PC** A program has a protected conversation with an OTMA client.
WAIT BID	Indicates that the OTMA client is processing a Client-Bid request.
WAITING	Indicates that the MPP region is waiting for work. Indicates a log gap is not being filled yet.

Table 195. Attributes and Status in the /DISPLAY Command (continued)

Status or Attribute	Meaning
WAIT RESPONSE	Indicates that the OTMA client is processing a message and is waiting for a response from the server.
WRTERR	Indicates a write I/O error on the OLDS or a database.

Appendix H. High Availability Large Database Commands

The following IMS commands can be issued against High Availability Large Databases (HALDBs). The commands are:

- /DBDUMP DATABASE
- /DBRECOVERY DATABASE
- /DISPLAY DATABASE
- /LOCK DATABASE
- /START DATABASE
- /STOP DATABASE
- QUERY DB
- /UNLOCK DATABASE
- UPDATE DB SET(LOCK(OFF))
- UPDATE DB SET(LOCK(ON))
- UPDATE DB START(ACCESS)
- UPDATE DB STOP(ACCESS)
- UPDATE DB STOP(SCHED)
- UPDATE DB STOP(UPDATES)

When you enter one of these commands, the database name can be an existing non-HALDB, a HALDB master, or a HALDB partition. A command against a HALDB partition operates exactly like a command against a non-HALDB with the exception of the /START DATABASE and the UPDATE DB START(ACCESS) command. A HALDB partition is not allocated during the command unless it was previously authorized but not allocated, the OPEN keyword was specified, or the partition has EEQEs. The partition is allocated at first reference.

The HALDB partition reflects conditions such as STOPPED, LOCKED, or NOTOPEN. When a HALDB partition is stopped, it must be explicitly started again. Commands with the keyword ALL and commands against a HALDB master do not change the STOPPED and LOCKED indicators in each HALDB partition.

When the command target is a HALDB master, processing acts on all HALDB partitions. For example, if the IMS command is /DBR on the HALDB master, all of the HALDB partitions are closed, deallocated, and unauthorized. Only the HALDB master displays STOPPED (each HALDB partition does not display STOPPED unless it was itself stopped). If a /DBR command was issued against a HALDB master, the display output of a /DISPLAY DATABASE command shows the HALDB master (as STOPPED), but does not display the status of the partitions.

Each partition inherits the access limitations of its HALDB master. If the /DBD command is issued against a HALDB master, all of its partitions close. A subsequent reference to any of the partitions results in the partition opening for input, although the partition's access might be UPDATE or EXCLUSIVE. The DBRC authorization state reflects the limited access.

© Copyright IBM Corp. 1974, 2003

Appendix I. IMS Commands, RACF Access Authorities and Resource Names Table

Table 196 applies to IMSplex command security. The table lists by IMS command verb and keyword the resource name and authorization that are used for RACF security checking.

For example, if you want to issue the `ACTIVATE NODE` command on an IMSplex named, PLX01, your userid must have UPDATE access to profile IMS.PLX01.ACT.NODE in the OPERCMDS class. If you only have READ authority, the `ACTIVATE NODE` command will fail with an 'insufficient authority' message.

In general, you can issue a display or query a command with READ authority. To change the state of a resource, you need UPDATE authority.

Note: You can reduce the number of RACF resources required by using wildcards in the resource name. For example, use IMS.PLX01.DIS.* to grant authority to use all display commands.

Table 196. Resource Names and RACF Authority for IMS Commands

Command Verb	Command Keyword	Authority	Resource Name
ACT	LINK	UPDATE	IMS.plxname.ACT.LINK
ACT	NODE	UPDATE	IMS.plxname.ACT.NODE
ALL	LU	UPDATE	IMS.plxname.ALL.LU
ASS	CLASS	UPDATE	IMS.plxname.ASS.CLASS
ASS	CPRI	UPDATE	IMS.plxname.ASS.CPRI
ASS	INPUT	UPDATE	IMS.plxname.ASS.INPUT
ASS	LCT	UPDATE	IMS.plxname.ASS.LCT
ASS	LPRI	UPDATE	IMS.plxname.ASS.LPRI
ASS	LTERM	UPDATE	IMS.plxname.ASS.LTERM
ASS	NPRI	UPDATE	IMS.plxname.ASS.NPRI
ASS	OUTPUT	UPDATE	IMS.plxname.ASS.OUTPUT
ASS	PARLIM	UPDATE	IMS.plxname.ASS.PARLIM
ASS	PLCT	UPDATE	IMS.plxname.ASS.PLCT
ASS	SEGNO	UPDATE	IMS.plxname.ASS.SEGNO
ASS	SEGSZ	UPDATE	IMS.plxname.ASS.SEGSZ
ASS	TRAN	UPDATE	IMS.plxname.ASS.TRAN
ASS	USER	UPDATE	IMS.plxname.ASS.USER
BRO	ACT	READ	IMS.plxname.BRO.ACT
BRO	LINE	READ	IMS.plxname.BRO.LINE
BRO	LTERM	READ	IMS.plxname.BRO.LTERM
BRO	MASTER	READ	IMS.plxname.BRO.MASTER
BRO	NODE	READ	IMS.plxname.BRO.NODE
BRO	PTERM	READ	IMS.plxname.BRO.PTERM
BRO	USER	READ	IMS.plxname.BRO.USER

© Copyright IBM Corp. 1974, 2003

Table 196. Resource Names and RACF Authority for IMS Commands (continued)

Command Verb	Command Keyword	Authority	Resource Name
CHA	APPC	UPDATE	IMS.plxname.CHA.APPC
CHA	CCTL	UPDATE	IMS.plxname.CHA.CCTL
CHA	CPLOG	UPDATE	IMS.plxname.CHA.CPLOG
CHA	DESC	UPDATE	IMS.plxname.CHA.DESC
CHA	DIR	UPDATE	IMS.plxname.CHA.DIR
CHA	FDR	UPDATE	IMS.plxname.CHA.FDR
CHA	LINK	UPDATE	IMS.plxname.CHA.LINK
CHA	NODE	UPDATE	IMS.plxname.CHA.NODE
CHA	PSWD	UPDATE	IMS.plxname.CHA.PSWD
CHA	SUBSYS	UPDATE	IMS.plxname.CHA.SUBSYS
CHA	SURV	UPDATE	IMS.plxname.CHA.SURV
CHA	TRAN	UPDATE	IMS.plxname.CHA.TRAN
CHA	UOR	UPDATE	IMS.plxname.CHA.UOR
CHA	USER	UPDATE	IMS.plxname.CHA.USER
CHE		UPDATE	IMS.plxname.CHE
CHE	DUMPQ	UPDATE	IMS.plxname.CHE.DUMPQ
CHE	FREEZE	UPDATE	IMS.plxname.CHE.FREEZE
CHE	PURGE	UPDATE	IMS.plxname.CHE.PURGE
CHE	STATISTICS	UPDATE	IMS.plxname.CHE.STATISTICS
CLS	NODE	UPDATE	IMS.plxname.CLS.NODE
CQC	SHRQ	UPDATE	IMS.plxname.CQC.SHRQ
CQC	SYSTEM	UPDATE	IMS.plxname.CQC.SYSTEM
CQQ	STATISTICS	READ	IMS.plxname.CQQ.STATISTICS
CQS	SHUTDOWN	UPDATE	IMS.plxname.CQS.SHUTDOWN
DBD	DB	UPDATE	IMS.plxname.DBD.DB
DBR	AREA	UPDATE	IMS.plxname.DBR.AREA
DBR	DB	UPDATE	IMS.plxname.DBR.DB
DBR	DATAGRP	UPDATE	IMS.plxname.DBR.DATAGRP
DEL	LE	UPDATE	IMS.plxname.DEL.LE
DEL	PSWD	UPDATE	IMS.plxname.DEL.PSWD
DEL	TERMINAL	UPDATE	IMS.plxname.DEL.TERMINAL
DEQ	AOITKN	UPDATE	IMS.plxname.DEQ.AOITKN
DEQ	LINE	UPDATE	IMS.plxname.DEQ.LINE
DEQ	LTERM	UPDATE	IMS.plxname.DEQ.LTERM
DEQ	LU	UPDATE	IMS.plxname.DEQ.LU
DEQ	MSNAME	UPDATE	IMS.plxname.DEQ.MSNAME
DEQ	NODE	UPDATE	IMS.plxname.DEQ.NODE
DEQ	SUSPEND	UPDATE	IMS.plxname.DEQ.SUSPEND
DEQ	TMEM	UPDATE	IMS.plxname.DEQ.TMEM

Table 196. Resource Names and RACF Authority for IMS Commands (continued)

Command Verb	Command Keyword	Authority	Resource Name
DEQ	TRAN	UPDATE	IMS.plxname.DEQ.TRAN
DEQ	USER	UPDATE	IMS.plxname.DEQ.USER
DIS	ACT	READ	IMS.plxname.DIS.ACT
DIS	AFFIN	READ	IMS.plxname.DIS.AFFIN
DIS	AOITKN	READ	IMS.plxname.DIS.AOITKN
DIS	APPC	READ	IMS.plxname.DIS.APPC
DIS	AREA	READ	IMS.plxname.DIS.AREA
DIS	ASMT	READ	IMS.plxname.DIS.ASMT
DIS	CCTL	READ	IMS.plxname.DIS.CCTL
DIS	CONV	READ	IMS.plxname.DIS.CONV
DIS	CPLOG	READ	IMS.plxname.DIS.CPLOG
DIS	CQS	READ	IMS.plxname.DIS.CQS
DIS	DB	READ	IMS.plxname.DIS.DB
DIS	DBD	READ	IMS.plxname.DIS.DBD
DIS	DESC	READ	IMS.plxname.DIS.DESC
DIS	FDR	READ	IMS.plxname.DIS.FDR
DIS	FPV	READ	IMS.plxname.DIS.FPV
DIS	HSB	READ	IMS.plxname.DIS.HSB
DIS	HSSP	READ	IMS.plxname.DIS.HSSP
DIS	LINE	READ	IMS.plxname.DIS.LINE
DIS	LINK	READ	IMS.plxname.DIS.LINK
DIS	LTERM	READ	IMS.plxname.DIS.LTERM
DIS	LU	READ	IMS.plxname.DIS.LU
DIS	MASTER	READ	IMS.plxname.DIS.MASTER
DIS	MODIFY	READ	IMS.plxname.DIS.MODIFY
DIS	MSNAME	READ	IMS.plxname.DIS.MSNAME
DIS	NODE	READ	IMS.plxname.DIS.NODE
DIS	OASN	READ	IMS.plxname.DIS.OASN
DIS	OLDS	READ	IMS.plxname.DIS.OLDS
DIS	OTMA	READ	IMS.plxname.DIS.OTMA
DIS	OVERFLOWQ	READ	IMS.plxname.DIS.OVERFLOWQ
DIS	POOL	READ	IMS.plxname.DIS.POOL
DIS	PGM	READ	IMS.plxname.DIS.PGM
DIS	PSB	READ	IMS.plxname.DIS.PSB
DIS	PTERM	READ	IMS.plxname.DIS.PTERM
DIS	Q	READ	IMS.plxname.DIS.Q
DIS	QCNT	READ	IMS.plxname.DIS.QCNT
DIS	RECOVERY	READ	IMS.plxname.DIS.RECOVERY
DIS	RTC	READ	IMS.plxname.DIS.RTC

Table 196. Resource Names and RACF Authority for IMS Commands (continued)

Command Verb	Command Keyword	Authority	Resource Name
DIS	SHUTDOWN	READ	IMS.plxname.DIS.SHUTDOWN
DIS	STATUS	READ	IMS.plxname.DIS.STATUS
DIS	STRUC	READ	IMS.plxname.DIS.STRUC
DIS	SUBSYS	READ	IMS.plxname.DIS.SUBSYS
DIS	SYSID	READ	IMS.plxname.DIS.SYSID
DIS	TIMEOVER	READ	IMS.plxname.DIS.TIMEOVER
DIS	TMEM	READ	IMS.plxname.DIS.TMEM
DIS	TRACE	READ	IMS.plxname.DIS.TRACE
DIS	TRACKING	READ	IMS.plxname.DIS.TRACKING
DIS	TRAN	READ	IMS.plxname.DIS.TRAN
DIS	UOR	READ	IMS.plxname.DIS.UOR
DIS	USER	READ	IMS.plxname.DIS.USER
END	LINE	UPDATE	IMS.plxname.END.LINE
END	NODE	UPDATE	IMS.plxname.END.NODE
END	USER	UPDATE	IMS.plxname.END.USER
EXC	LINE	UPDATE	IMS.plxname.EXC.LINE
EXC	NODE	UPDATE	IMS.plxname.EXC.NODE
EXC	USER	UPDATE	IMS.plxname.EXC.USER
ERE		UPDATE	IMS.plxname.ERE
ERE	BACKUP	UPDATE	IMS.plxname.ERE.BACKUP
ERE	COLDBASE	UPDATE	IMS.plxname.ERE.COLDBASE
ERE	COLDCOMM	UPDATE	IMS.plxname.ERE.COLDCOMM
ERE	COLDSYS	UPDATE	IMS.plxname.ERE.COLDSYS
EXI	CONV	UPDATE	IMS.plxname.EXI.CONV
IDL	LINE	UPDATE	IMS.plxname.IDL.LINE
IDL	LINK	UPDATE	IMS.plxname.IDL.LINK
IDL	NODE	UPDATE	IMS.plxname.IDL.NODE
INIT	OLC	UPDATE	IMS.plxname.INIT.OLC
LOC	DB	UPDATE	IMS.plxname.LOC.DB
LOC	PGM	UPDATE	IMS.plxname.LOC.PGM
LOC	TRAN	UPDATE	IMS.plxname.LOC.TRAN
LOG		UPDATE	IMS.plxname.LOG
MOD	ABORT	UPDATE	IMS.plxname.MOD.ABORT
MOD	COMMIT	UPDATE	IMS.plxname.MOD.COMMIT
MOD	PREPARE	UPDATE	IMS.plxname.MOD.PREPARE
MON	LINE	UPDATE	IMS.plxname.MON.LINE
MSA	LINK	UPDATE	IMS.plxname.MSA.LINK
MSA	MSNAME	UPDATE	IMS.plxname.MSA.MSNAME
MSA	SYSID	UPDATE	IMS.plxname.MSA.SYSID

Table 196. Resource Names and RACF Authority for IMS Commands (continued)

Command Verb	Command Keyword	Authority	Resource Name
MSA	TRAN	UPDATE	IMS.plxname.MSA.TRAN
MSV	MSNAME	UPDATE	IMS.plxname.MSV.MSNAME
MSV	SYSID	UPDATE	IMS.plxname.MSV.SYSID
NRE		UPDATE	IMS.plxname.NRE
NRE	CHKPT	UPDATE	IMS.plxname.NRE.CHKPT
OPN	NODE	UPDATE	IMS.plxname.OPN.NODE
PST	LINE	UPDATE	IMS.plxname.PST.LINE
PST	LINK	UPDATE	IMS.plxname.PST.LINK
PST	LTERM	UPDATE	IMS.plxname.PST.LTERM
PST	MSPLINK	UPDATE	IMS.plxname.PST.MSPLINK
PST	REGION	UPDATE	IMS.plxname.PST.REGION
PST	TRAN	UPDATE	IMS.plxname.PST.TRAN
PUR	APPC	UPDATE	IMS.plxname.PUR.APPC
PUR	FPPROG	UPDATE	IMS.plxname.PUR.FPPROG
PUR	FPRGN	UPDATE	IMS.plxname.PUR.FPRGN
PUR	LINE	UPDATE	IMS.plxname.PUR.LINE
PUR	LTERM	UPDATE	IMS.plxname.PUR.LTERM
PUR	MSNAME	UPDATE	IMS.plxname.PUR.MSNAME
PUR	TRAN	UPDATE	IMS.plxname.PUR.TRAN
QRY	IMSPLEX	READ	IMS.plxname.QRY.IMSPLEX
QRY	LE	READ	IMS.plxname.QRY.LE
QRY	MEMBER	READ	IMS.plxname.QRY.MEMBER
QRY	OLC	READ	IMS.plxname.QRY.OLC
QRY	STRUCTURE	READ	IMS.plxname.QRY.STRUCTURE
QRY	TRAN	READ	IMS.plxname.QRY.TRAN
QUI	NODE	UPDATE	IMS.plxname.QUI.NODE
RDI	MASTER	READ	IMS.plxname.RDI.MASTER
REC	ADD	UPDATE	IMS.plxname.REC.ADD
REC	REMOVE	UPDATE	IMS.plxname.REC.REMOVE
REC	START	UPDATE	IMS.plxname.REC.START
REC	STOP	UPDATE	IMS.plxname.REC.STOP
REC	TERMINATE	UPDATE	IMS.plxname.REC.TERMINATE
RMC		UPDATE	IMS.plxname.RMC
RMD		UPDATE	IMS.plxname.RMD
RMG		UPDATE	IMS.plxname.RMG
RMI		UPDATE	IMS.plxname.RMI
RML		READ	IMS.plxname.RML
RMN		UPDATE	IMS.plxname.RMN
RST	LINE	UPDATE	IMS.plxname.RST.LINE

Table 196. Resource Names and RACF Authority for IMS Commands (continued)

Command Verb	Command Keyword	Authority	Resource Name
RST	LINK	UPDATE	IMS.plxname.RST.LINK
RST	MSPLINK	UPDATE	IMS.plxname.RST.MSPLINK
RST	NODE	UPDATE	IMS.plxname.RST.NODE
RST	USER	UPDATE	IMS.plxname.RST.USER
RTA	DUMPQ	UPDATE	IMS.plxname.RTA.DUMPQ
RTA	FREEZE	UPDATE	IMS.plxname.RTA.FREEZE
RTA	UNPLAN	UPDATE	IMS.plxname.RTA.UNPLAN
SEC	APPC	UPDATE	IMS.plxname.SEC.APPC
SEC	OTMA	UPDATE	IMS.plxname.SEC.OTMA
SMC	MASTER	UPDATE	IMS.plxname.SMC.MASTER
SMC	TERMINAL	UPDATE	IMS.plxname.SMC.TERMINAL
STA	APPC	UPDATE	IMS.plxname.STA.APPC
STA	AREA	UPDATE	IMS.plxname.STA.AREA
STA	AUTOARCH	UPDATE	IMS.plxname.STA.AUTOARCH
STA	CLASS	UPDATE	IMS.plxname.STA.CLASS
STA	DB	UPDATE	IMS.plxname.STA.DB
STA	DATAGRP	UPDATE	IMS.plxname.STA.DATAGRP
STA	DC	UPDATE	IMS.plxname.STA.DC
STA	ISOLOG	UPDATE	IMS.plxname.STA.ISOLOG
STA	LINE	UPDATE	IMS.plxname.STA.LINE
STA	LTERM	UPDATE	IMS.plxname.STA.LTERM
STA	LU	UPDATE	IMS.plxname.STA.LU
STA	MADSIOT	UPDATE	IMS.plxname.STA.MADSIOT
STA	MSNAME	UPDATE	IMS.plxname.STA.MSNAME
STA	NODE	UPDATE	IMS.plxname.STA.NODE
STA	OLDS	UPDATE	IMS.plxname.STA.OLDS
STA	OTMA	UPDATE	IMS.plxname.STA.OTMA
STA	PGM	UPDATE	IMS.plxname.STA.PGM
STA	REGION	UPDATE	IMS.plxname.STA.REGION
STA	RTC	UPDATE	IMS.plxname.STA.RTC
STA	SB	UPDATE	IMS.plxname.STA.SB
STA	SERVGRP	UPDATE	IMS.plxname.STA.SERVGRP
STA	SUBSYS	UPDATE	IMS.plxname.STA.SUBSYS
STA	SURV	UPDATE	IMS.plxname.STA.SURV
STA	THREAD	UPDATE	IMS.plxname.STA.THREAD
STA	TMEM	UPDATE	IMS.plxname.STA.TMEM
STA	TRAN	UPDATE	IMS.plxname.STA.TRAN
STA	TRKARCH	UPDATE	IMS.plxname.STA.TRKARCH
STA	USER	UPDATE	IMS.plxname.STA.USER

Table 196. Resource Names and RACF Authority for IMS Commands (continued)

Command Verb	Command Keyword	Authority	Resource Name
STA	VGR	UPDATE	IMS.plxname.STA.VGR
STA	WADS	UPDATE	IMS.plxname.STA.WADS
STA	XRCTRACK	UPDATE	IMS.plxname.STA.XRCTRACK
STO	ADS	UPDATE	IMS.plxname.STO.ADS
STO	APPC	UPDATE	IMS.plxname.STO.APPC
STO	AREA	UPDATE	IMS.plxname.STO.AREA
STO	AUTOARCH	UPDATE	IMS.plxname.STO.AUTOARCH
STO	BACKUP	UPDATE	IMS.plxname.STO.BACKUP
STO	CLASS	UPDATE	IMS.plxname.STO.CLASS
STO	DB	UPDATE	IMS.plxname.STO.DB
STO	DATAGRP	UPDATE	IMS.plxname.STO.DATAGRP
STO	DC	UPDATE	IMS.plxname.STO.DC
STO	LINE	UPDATE	IMS.plxname.STO.LINE
STO	LTERM	UPDATE	IMS.plxname.STO.LTERM
STO	LU	UPDATE	IMS.plxname.STO.LU
STO	MADSIOT	UPDATE	IMS.plxname.STO.MADSIOT
STO	MSNAME	UPDATE	IMS.plxname.STO.MSNAME
STO	NODE	UPDATE	IMS.plxname.STO.NODE
STO	OLDS	UPDATE	IMS.plxname.STO.OLDS
STO	OTMA	UPDATE	IMS.plxname.STO.OTMA
STO	PGM	UPDATE	IMS.plxname.STO.PGM
STO	REGION	UPDATE	IMS.plxname.STO.REGION
STO	RTC	UPDATE	IMS.plxname.STO.RTC
STO	SB	UPDATE	IMS.plxname.STO.SB
STO	SERVGRP	UPDATE	IMS.plxname.STO.SERVGRP
STO	SUBSYS	UPDATE	IMS.plxname.STO.SUBSYS
STO	SURV	UPDATE	IMS.plxname.STO.SURV
STO	THREAD	UPDATE	IMS.plxname.STO.THREAD
STO	TMEM	UPDATE	IMS.plxname.STO.TMEM
STO	TRAN	UPDATE	IMS.plxname.STO.TRAN
STO	USER	UPDATE	IMS.plxname.STO.USER
STO	VGR	UPDATE	IMS.plxname.STO.VGR
STO	WADS	UPDATE	IMS.plxname.STO.WADS
STO	XRCTRACK	UPDATE	IMS.plxname.STO.XRCTRACK
SWI	OLDS	UPDATE	IMS.plxname.SWI.OLDS
SWI	SYSTEM	UPDATE	IMS.plxname.SWI.SYSTEM
SWI	WADS	UPDATE	IMS.plxname.SWI.WADS
TERM	OLC	UPDATE	IMS.plxname.TERM.OLC
TES	MFS	UPDATE	IMS.plxname.TES.MFS

Table 196. Resource Names and RACF Authority for IMS Commands (continued)

Command Verb	Command Keyword	Authority	Resource Name
TRA	SET	UPDATE	IMS.plxname.TRA.SET
UNL	DB	UPDATE	IMS.plxname.UNL.DB
UNL	PGM	UPDATE	IMS.plxname.UNL.PGM
UNL	SYSTEM	UPDATE	IMS.plxname.UNL.SYSTEM
UNL	TRAN	UPDATE	IMS.plxname.UNL.TRAN
UPD	LE	UPDATE	IMS.plxname.UPD.LE
UPD	TRAN	UPDATE	IMS.plxname.UPD.TRAN
VUN	AREA	UPDATE	IMS.plxname.VUN.AREA

Bibliography

This bibliography includes all the publications cited in this book, including the publications in the IMS library.

IMS Queue Control Facility for z/OS: User's Guide, V1 R2, SC26-9685

MVS/ESA Operations: JES2 Commands, SC28-1039

MVS/ESA Operations: JES3 Commands, SC23-0074

MVS/ESA System Commands, GC28-1442

OS/390 MVS System Commands, GC28-1781

RACF Macros and Interfaces , SC23-3732

RACF Security Administrator's Guide, SC23-3726

IMS Version 9 Library

SC27-1283	ADB	*IMS Version 9: Administration Guide: Database Manager*
SC27-1284	AS	*IMS Version 9: Administration Guide: System*
SC27-1285	ATM	*IMS Version 9: Administration Guide: Transaction Manager*
SC27-1286	APDB	*IMS Version 9: Application Programming: Database Manager*
SC27-1287	APDG	*IMS Version 9: Application Programming: Design Guide*
SC27-1288	APCICS	*IMS Version 9: Application Programming: EXEC DLI Commands for CICS and IMS*
SC27-1289	APTM	*IMS Version 9: Application Programming: Transaction Manager*
SC27-1290	BPE	*IMS Version 9: Base Primitive Environment Guide and Reference*
SC27-1291	CR	*IMS Version 9: Command Reference*
SC27-1292	CQS	*IMS Version 9: Common Queue Server Guide and Reference*
SC27-1293	CSL	*IMS Version 9: Common Service Layer Guide and Reference*
SC27-1294	CG	*IMS Version 9: Customization Guide*
SC27-1295	DBRC	*IMS Version 9: DBRC Guide and Reference*
LY37-3742	DGR	*IMS Version 9: Diagnosis Guide and Reference*
LY37-3743	FAST	*IMS Version 9: Failure Analysis Structure Tables (FAST) for Dump Analysis*
SC27-1296	IJUG	*IMS Version 9: IMS Java Guide and Reference*
GC27-1297	IIV	*IMS Version 9: Installation Volume 1: Installation Verification*
GC27-1298	ISDT	*IMS Version 9: Installation Volume 2: System Definition and Tailoring*
SC27-1300	MIG	*IMS Version 9: Master Index and Glossary*
GC27-1301	MC1	*IMS Version 9: Messages and Codes, Volume 1*
GC27-1302	MC2	*IMS Version 9: Messages and Codes, Volume 2*
SC27-1303	OTMA	*IMS Version 9: Open Transaction Manager Access Guide*
SC27-1304	OG	*IMS Version 9: Operations Guide*
GC27-1305	RPG	*IMS Version 9: Release Planning Guide*
SC27-1308	URDBTM	*IMS Version 9: Utilities Reference: Database and Transaction Manager*
SC27-1309	URS	*IMS Version 9: Utilities Reference: System*

Supplementary Publications

GC27-1299	LPS	*IMS Version 9: Licensed Program Specifications*
SC27-1307	SOC	*IMS Version 9: Summary of Operator Commands*

Publication Collections

LK3T-7092	CD	IMS Version 9 Softcopy Library
LBOF-7780	Hardcopy and CD	Licensed Bill of Forms (LBOF): IMS Version 9 Hardcopy and Softcopy Library
SBOF-7781	Hardcopy	Unlicensed Bill of Forms (SBOF): IMS Version 9 Unlicensed Hardcopy Library
SK2T-6700	CD	OS/390 Collection
SK3T-4270	CD	z/OS Software Products Collection

Accessibility Titles Cited in this Book

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

© Copyright IBM Corp. 1974, 2003

Program Number: 5655-J38

IBM Confidential
Printed in USA

ZES1-2356-00