

1.6

*IBM Operations Manager for z/VM  
Administration Guide*



**September, 2022**

This edition applies to Version 1 Release 6 of IBM Operations Manager for z/VM (product number 5697-J10) and to all subsequent releases and modifications until otherwise indicated in new editions.

**© Copyright IBM Corporation 1994, 2022; Copyright Rocket Software 2006, 2022**

**© Copyright International Business Machines Corporation .**

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

---

# Contents

<b>Figures.....</b>	<b>xiii</b>
<b>Tables.....</b>	<b>xix</b>
<b>About this information.....</b>	<b>xxv</b>
Service updates and Support information.....	xxv
<b>Summary of Changes.....</b>	<b>xxvii</b>
<b>Chapter 1. Introduction to Operations Manager.....</b>	<b>1</b>
What does Operations Manager do?.....	1
Benefits of using Operations Manager.....	1
Operations Manager components.....	2
Message processing component.....	2
Action processing component.....	2
User interaction component.....	2
Automated session component.....	2
Component interaction.....	2
Accessing Help.....	3
Prerequisites.....	3
Hardware requirements.....	3
Software requirements.....	3
<b>Chapter 2. Configuring Operations Manager.....</b>	<b>5</b>
Step 1. Update OPMGRM1.....	5
Updates to PROFILE EXEC .....	5
Updates to CP directory.....	6
Step 2. Update Action Processing Servers.....	7
Updates to PROFILE EXEC.....	7
Updates to CP directory.....	8
Step 3. Update End User and Administrator IDs.....	8
Step 4. Update IDs monitored by Operations Manager.....	9
Updating the CP directory.....	9
Updating the PROFILE EXEC.....	9
Verifying the console data is being sent to Operations Manager.....	10
Step 5. (Optional) Configure an external security mechanism.....	10
Set up example for IBM Resource Access Control Facility (RACF).....	10
ESM class and Facility class profiles.....	11
<b>Chapter 3. Creating a startup configuration file.....</b>	<b>21</b>
Using the sample configuration file.....	21
Copying the sample configuration file.....	21
Adding comments.....	22
Assigning authority.....	22
Control authority.....	23
Configuration authority.....	23
Console access.....	23
Spool access.....	23
IP session interaction.....	24

Authority assignments with AUTH and external security.....	24
Defining action processing servers.....	24
Defining system options.....	24
<b>Chapter 4. Starting Operations Manager and managing the configuration file.....</b>	<b>27</b>
Modifying the startup configuration file.....	27
Start the Operations Manager service machines.....	27
Creating shortcuts to common commands.....	28
Verify the installation and configuration.....	28
Managing the configuration file.....	28
<b>Chapter 5. Entering commands.....</b>	<b>31</b>
Command syntax rules.....	31
Where to enter commands.....	32
<b>Chapter 6. Syntax for System Management Commands.....</b>	<b>35</b>
Purpose of system management commands.....	35
GOMMAIN.....	35
Authorization.....	35
Purpose.....	35
Operands.....	35
Usage notes.....	36
Return codes.....	36
GOMCMD.....	36
Authorization.....	36
Purpose.....	37
Operands.....	37
Usage notes.....	37
Examples.....	37
Return codes.....	37
GOMSVM.....	38
Authorization.....	38
Purpose.....	38
Operands.....	38
Usage notes.....	38
Return codes.....	39
GOMTDM.....	39
Authorization.....	40
Purpose.....	40
Operands.....	40
Usage notes.....	41
Return codes.....	42
<b>Chapter 7. Syntax for control commands.....</b>	<b>43</b>
Purpose of control commands.....	43
Authorization for control commands.....	43
AUTH.....	43
Authorization.....	43
Purpose.....	43
Operands.....	43
Operand value lengths.....	44
Examples.....	44
Usage notes.....	45
Return codes.....	45
CMS.....	45
Authorization.....	45
Purpose.....	45

Operands.....	46
Operand value lengths.....	46
Examples.....	46
Usage notes.....	46
Return codes.....	46
CONFIG.....	46
Authorization.....	47
Purpose.....	47
Operands.....	47
Operand value lengths.....	47
Example.....	48
Usage notes.....	48
Return codes.....	48
CP.....	48
Authorization.....	49
Purpose.....	49
Operands.....	49
Operand value lengths.....	49
Examples.....	49
Usage notes.....	49
Return codes.....	50
DEFDSPS.....	50
Authorization.....	50
Purpose.....	50
Operands.....	50
Operand value lengths.....	51
Examples.....	51
Usage notes.....	51
Return codes.....	52
DEFIPCS.....	52
Authority.....	52
Purpose.....	52
Operands.....	52
Operand value lengths.....	53
Examples.....	53
Usage notes.....	53
Return codes.....	56
DEFOPTN.....	57
Authorization.....	57
Purpose.....	57
Operands.....	57
Operand value lengths.....	61
Example.....	62
Usage notes.....	62
Return codes.....	62
DEFSRVM.....	63
Authorization.....	63
Purpose.....	63
Operands.....	63
Operand value lengths.....	63
Example.....	64
Usage notes.....	64
Return codes.....	64
DEFVIEW.....	64
Authorization.....	64
Purpose.....	64
Operands.....	64
Operand value lengths.....	65

Examples.....	65
Usage notes.....	66
Return codes.....	66
DELACTION.....	66
Authorization.....	66
Purpose.....	66
Operands.....	66
Operand value lengths.....	66
Examples.....	67
Usage notes.....	67
Return codes.....	67
DELDSPS.....	68
Authorization.....	68
Purpose.....	68
Operands.....	68
Operand value lengths.....	68
Examples.....	68
Usage notes.....	69
Return codes.....	69
DELIPCS.....	69
Authority.....	69
Purpose.....	69
Operands.....	69
Operand value lengths.....	69
Example.....	70
Usage notes.....	70
DELSRVM.....	70
Authorization.....	70
Purpose.....	70
Operands.....	70
Operand value lengths.....	70
Example.....	71
Usage notes.....	71
Return codes.....	71
DELVIEW.....	71
Authorization.....	71
Purpose.....	71
Operands.....	71
Operand value lengths.....	72
Examples.....	72
Usage notes.....	72
Return codes.....	73
QUIT.....	73
Authorization.....	73
Purpose.....	73
Example.....	73
Usage notes.....	73
Return codes.....	73
REVOKE.....	74
Authorization.....	74
Purpose.....	74
Operands.....	74
Operand value lengths.....	74
Example.....	75
Usage notes.....	75
Return codes.....	75
STATUS.....	76
Authorization.....	76

Purpose.....	76
Operands.....	76
Operand value lengths.....	77
Examples.....	77
Usage notes.....	79
Return codes.....	79
STOP.....	80
Authorization.....	80
Purpose.....	80
Example.....	80
Usage note.....	80
Return codes.....	80

## **Chapter 8. Syntax for Configuration Commands.....81**

Purpose of configuration commands.....	81
Authorization for configuration commands.....	81
CONFIG.....	81
Authorization.....	82
Purpose.....	82
Operands.....	82
Operand value lengths.....	82
Example.....	83
Usage notes.....	83
Return codes.....	83
DEFACTN.....	84
Authorization.....	85
Purpose.....	85
Operands.....	85
Example.....	87
Operand value lengths.....	88
Usage notes.....	88
Return codes.....	89
DEFDATE.....	89
Authorization.....	89
Purpose.....	89
Operands.....	89
Operand value lengths.....	90
Example.....	90
Usage notes.....	90
Return codes.....	90
DEFEMON.....	91
Authorization.....	91
Purpose.....	91
Operands.....	91
Operand value lengths.....	93
Examples.....	93
Usage notes.....	94
Return codes.....	94
DEFIMON.....	95
Authorization .....	95
Purpose.....	95
Operands.....	95
Operand value lengths.....	97
Example.....	97
Usage notes.....	98
Return codes.....	98
DEFGROUP.....	99

Authorization.....	99
Purpose.....	99
Operands.....	99
Operand value lengths.....	100
Example.....	100
Usage notes.....	100
Return codes.....	100
DEFMMON.....	101
Authorization.....	101
Purpose.....	101
Operands.....	101
Operand value lengths.....	103
Example.....	104
Usage notes.....	104
Return codes.....	105
DEFPMON.....	105
Authorization.....	105
Purpose.....	105
Operands.....	106
Operand value lengths.....	107
Example.....	108
Usage notes.....	108
Return codes.....	109
DEFRULE.....	109
Authorization.....	109
Purpose.....	110
Operands.....	110
Operand value lengths.....	112
Example.....	112
Usage notes.....	114
Return Codes.....	115
DEFSCHD.....	115
Authorization.....	116
Purpose.....	116
Operands.....	116
Operand value lengths.....	118
Examples.....	119
Usage notes.....	119
Return codes.....	119
DEFSMON.....	120
Authorization.....	120
Purpose.....	120
Operands.....	120
Operand value lengths.....	122
Example.....	122
Usage notes.....	123
Return codes.....	123
DELACTION.....	124
Authorization.....	124
Purpose.....	124
Operands.....	124
Operand value lengths.....	124
Example.....	124
Usage notes.....	124
Return codes.....	124
DELDATE.....	125
Authorization.....	125
Purpose.....	125



Operands.....	125
Operand value lengths.....	125
Example.....	125
Usage notes.....	125
Return codes.....	125
DELEMON.....	126
Authorization.....	126
Purpose.....	126
Operands.....	126
Operand value lengths.....	126
Example.....	126
Usage notes.....	126
Return codes.....	126
DELGROUP.....	127
Authorization.....	127
Purpose.....	127
Operands.....	127
Operand value lengths.....	127
Example.....	128
Usage notes.....	128
Return codes.....	128
DELIMON.....	128
Authorization.....	128
Purpose.....	128
Operands.....	128
Operand value lengths.....	128
Example.....	129
Usage notes.....	129
Return codes.....	129
DELMON.....	129
Authorization.....	129
Purpose.....	129
Operands.....	129
Operand value lengths.....	129
Example.....	130
Usage note.....	130
Return codes.....	130
DELPMON.....	130
Authorization.....	130
Purpose.....	130
Operands.....	130
Operand value lengths.....	130
Example.....	131
Usage notes.....	131
Return codes.....	131
DELRULE.....	131
Authorization.....	131
Purpose.....	131
Operands.....	131
Operand value lengths.....	131
Example.....	132
Usage notes.....	132
Return Codes.....	132
DELSCHD.....	132
Authorization.....	132
Purpose.....	132
Operands.....	132
Operand value lengths.....	132

Example.....	133
Usage notes.....	133
Return codes.....	133
DELSMON.....	133
Authorization.....	133
Purpose.....	133
Operands.....	133
Operand value lengths.....	133
Example.....	134
Usage notes.....	134
Return codes.....	134
RESUME.....	134
Authorization.....	134
Purpose.....	134
Operands.....	135
Operand value lengths.....	135
Example.....	136
Usage notes.....	136
Return codes.....	136
RUN.....	137
Authorization.....	137
Purpose.....	137
Operands.....	137
Operand value lengths.....	138
Examples.....	138
Usage notes.....	139
Return codes.....	139
STATUS.....	139
Authorization.....	139
Purpose.....	139
Operands.....	139
Operand value lengths.....	140
Examples.....	140
Usage notes.....	143
Return codes.....	143
SUSPEND.....	143
Authorization.....	143
Purpose.....	144
Operands.....	144
Operand value lengths.....	144
Example.....	145
Usage notes.....	145
Return codes.....	145

## Chapter 9. Syntax for console, log, and spool access commands and commands

<b>for IP session interaction.....</b>	<b>147</b>
Purpose of console, log, and spool access commands.....	147
Authorization for console, log, and spool access commands.....	147
Console data spaces.....	147
ALTRCON.....	148
Authority.....	148
Purpose.....	148
Operands.....	148
Operand value lengths.....	149
Examples.....	149
Usage notes.....	150
Return codes.....	150

CMDIPCS.....	150
Authority.....	150
Purpose.....	150
Operands.....	151
Example.....	151
Usage notes.....	151
LOGTEXT.....	151
Authorization.....	152
Purpose.....	152
Operands.....	152
Operand value lengths.....	152
Example.....	152
Usage notes.....	152
Return codes.....	152
VIEWCON.....	153
Authorization.....	153
Purpose.....	153
Operands.....	153
Operand value lengths.....	154
Examples.....	155
Usage notes.....	156
Return codes.....	159
VIEWLOG.....	159
Authorization.....	159
Purpose.....	160
Operands.....	160
Operand value lengths.....	160
Examples.....	161
Usage notes.....	162
Return codes.....	165
IEWSPL.....	165
Authorization.....	165
Purpose.....	166
Operands.....	166
Operand value lengths.....	166
Examples.....	167
Usage notes.....	171
Return codes.....	176
<b>Appendix A. Sample files.....</b>	<b>177</b>
<b>Appendix B. Wildcard pattern matching.....</b>	<b>181</b>
<b>Appendix C. Using substitution variables in commands.....</b>	<b>183</b>
<b>Appendix D. Accounting and Monitoring Information.....</b>	<b>193</b>
Accounting record format.....	193
Monitoring record format.....	194
<b>Appendix E. Log File Management.....</b>	<b>195</b>
Customize log file management.....	195
<b>Appendix F. Using internet protocol and networked components.....</b>	<b>197</b>
Domain Name System (DSN) name resolution.....	197
Resolving a DSN name.....	197
Using multiple DSN resolvers.....	197

Sharing an action processing environment.....	198
Using processing environment of another instance.....	198
Sharing processing environment of multiple instances.....	199
Collection and processing of syslog data.....	200
Translation tables.....	201
Collection and processing of remote file input.....	202
Send and view file content.....	202
Send and view multiple files.....	202
GOMRSIF.....	203
Authorization.....	203
Purpose.....	203
Operands.....	203
Usage notes.....	204
Return codes.....	204
<b>Appendix G. Testing Operations Manager installation and configuration.....</b>	<b>205</b>
Checking the status of OPMGRM1.....	205
Verifying consoles are monitored and viewable.....	205
Creating a schedule and corresponding action.....	206
Verifying spool files are viewable.....	206
<b>Appendix H. Application and settings interference.....</b>	<b>207</b>
<b>Appendix I. GOMGLBL facility.....</b>	<b>209</b>
Preparing GOMGLBL.....	209
GOMGLBL.....	210
Authorization.....	210
Purpose.....	210
Operands.....	210
Usage notes.....	211
Restrictions.....	211
Return codes.....	211
<b>Appendix J. How to read syntax diagrams.....</b>	<b>213</b>
<b>Appendix K. Messages and Codes.....</b>	<b>215</b>
<b>Notices.....</b>	<b>263</b>
Trademarks.....	264
Terms and conditions for product documentation.....	264
Privacy policy considerations.....	265
<b>Index.....</b>	<b>267</b>

---

# Figures

1. Operations Manager component interaction.....	3
2. GOMMAIN command syntax.....	35
3. GOMCMD command syntax.....	36
4. GOMSVM command syntax.....	38
5. GOMTDM command syntax.....	40
6. AUTH command syntax.....	43
7. AUTH command with GROUP SYSADM operands.....	44
8. AUTH command with USER operand.....	45
9. CMS command syntax.....	45
10. Example of CMS command .....	46
11. CONFIG command syntax.....	47
12. CP command syntax.....	48
13. Example of CP command with USERID operand.....	49
14. Example of CP command with CPLEVEL operand .....	49
15. Example of CP command with USERS operand.....	49
16. DEFDSPS command syntax.....	50
17. DEFDSPS command issued for user.....	51
18. DEFDSPS command issued for group.....	51
19. DEFIPCS command syntax.....	52
20. DEFOPTN command syntax.....	57
21. DEFOPTN command.....	62
22. DEFSRVM command syntax.....	63
23. DEFSRVM commands in config file.....	64

24. DEFVIEW command syntax.....	64
25. DEFVIEW command with two users.....	65
26. VIEWCON command.....	65
27. DEFGROUP commands combined with DEFVIEW.....	65
28. DELACTQ command syntax.....	66
29. List of actions before the DELACTQ command is issued.....	67
30. DELACTQ command issued.....	67
31. List of actions after the DELACTQ command is issued.....	67
32. DELDPS command syntax.....	68
33. DELDPS command issued for user MAINT640.....	68
34. DELDPS command issued for group ROCKET1.....	69
35. DELIPCS command syntax.....	69
36. DELIPCS command.....	70
37. DELSRVM command syntax.....	70
38. DELSRVM command.....	71
39. DELVIEW command syntax.....	71
40. DELVIEW command to delete a user.....	72
41. DELVIEW command to delete a group.....	72
42. QUIT command syntax.....	73
43. QUIT command issued.....	73
44. REVOKE command syntax.....	74
45. REVOKE command issued.....	75
46. "Not authorized" message.....	75
47. STATUS command syntax.....	76
48. Summary output using STATUS command without operands .....	77

49. STATUS command with DETAIL AUTH operand.....	78
50. STATUS command with DETAIL CONSOLE operands.....	78
51. Details about monitored virtual machines.....	79
52. Details about all spool usage monitors.....	79
53. Details about users' spool files.....	79
54. STOP command syntax.....	80
55. STOP command issued.....	80
56. CONFIG command syntax.....	82
57. DEFACTN command syntax.....	85
58. Example of DEFACTN command with DEFSCHD "Now".....	87
59. Example of DEFACTN command with DEFMMON.....	87
60. DEFDATE command syntax.....	89
61. DEFDATE command issued.....	90
62. DEFEMON command syntax.....	91
63. Example of DEFEMON command with DEFACTN command.....	93
64. DEFIMON command syntax.....	95
65. Example of DEFIMON command with DEFACTN command.....	98
66. DEFGROUP command syntax.....	99
67. DEFGROUP command issued.....	100
68. DEFMMON command syntax.....	101
69. DEFACTN and DEFMMON commands.....	104
70. DEFPMON command syntax.....	105
71. Example of DEFPMON command with DEFACTN command.....	108
72. DEFRULE command syntax.....	109
73. DEFRULE command issued with DEFACTN command.....	113

74. Message displayed on console.....	113
75. DEFSCHD command syntax.....	116
76. DEFSCHD command issued.....	119
77. DEFSMON command syntax.....	120
78. Example of DEFSMON command with DEFACTN command.....	122
79. DELACTN command syntax.....	124
80. DELACTN command issued.....	124
81. DELDATE command syntax.....	125
82. DELDATE command issued.....	125
83. DELEMON command syntax.....	126
84. DELEMON command issued.....	126
85. DELGROUP command syntax.....	127
86. DELGROUP command issued.....	128
87. DELIMON command syntax.....	128
88. DELIMON command issued.....	129
89. DELMMON command syntax.....	129
90. DELMMON command.....	130
91. DELPMON command syntax.....	130
92. DELPMON command.....	131
93. DELRULE command syntax.....	131
94. DELRULE command issued.....	132
95. DELSCHD command syntax.....	132
96. DELSCHD command.....	133
97. DELSMON command syntax.....	133
98. DELSMON command issued.....	134



99. RESUME command syntax.....	134
100. RESUME command issued.....	136
101. RUN command syntax.....	137
102. RUN command for RSAMPLE action.....	138
103. Run command with CP message.....	139
104. STATUS command syntax.....	139
105. Summary output using STATUS command without operands .....	141
106. STATUS command with DETAIL AUTH operand.....	141
107. STATUS command with DETAIL CONSOLE operands.....	142
108. Details about monitored virtual machines.....	142
109. Details about all spool usage monitors.....	142
110. Details about users' spool files.....	143
111. SUSPEND command syntax.....	143
112. SUSPEND command issued.....	145
113. ALTRCON command syntax .....	148
114. ALTRCON command issued.....	150
115. CMDIPCS command syntax.....	150
116. LOGTEXT command syntax.....	152
117. VIEWCON command syntax.....	153
118. VIEWCON command.....	155
119. VIEWCON command.....	155
120. Example of DEFVIEW for multiple user ID's.....	155
121. Example of VIEWCON for combined view.....	155
122. VIEWLOG command syntax.....	159
123. VIEWLOG command without operands.....	161

124. VIEWLOG command with LINES operand.....	162
125. VIEWLOG command with DATE operand.....	162
126. VIEWSPL command syntax.....	165
127. VIEWSPL with no operands.....	167
128. VIEWSPL command with USER operand.....	168
129. VIEWSPL command with DATE operand.....	168
130. VIEWSPL command with SIZE operand.....	169
131. VIEWSPL command with SFID USER,SPLNBR operands.....	170
132. VIEWSPL command with MODE RDR operand.....	170
133. VIEWSPL command with several operands.....	171
134. GOMRSIF command syntax.....	203
135. GOMGLBL command syntax.....	210

---

# Tables

1. General ESMCLASS profiles.....	12
2. Command-level ESMCLASS profiles.....	13
3. Command-level profile extensions.....	18
4. GOMMAIN return codes.....	36
5. GOMCMD return codes.....	38
6. GOMSVM return codes.....	39
7. GOMTDM return codes.....	42
8. AUTH operand value lengths.....	44
9. AUTH return codes.....	45
10. CMS operand value lengths.....	46
11. CMS return codes.....	46
12. CONFIG operand value lengths.....	47
13. CONFIG return codes.....	48
14. CP operand value lengths.....	49
15. CP return codes.....	50
16. DEFDSPS operand value lengths.....	51
17. DEFDSPS return codes.....	52
18. DEFIPCS operand value lengths.....	53
19. PARM Operand fields.....	54
20. PARM Operand fields.....	54
21. PARM Operand fields.....	55
22. PARM Operand fields.....	56
23. PARM Operand fields.....	56

24. DEFIPCS return codes.....	56
25. Extended command levels.....	59
26. DEFOPTN operand value lengths.....	61
27. DEFOPTN return codes.....	62
28. DEFSRVM operand value lengths.....	63
29. DEFSRVM return codes.....	64
30. DEFVIEW operand value lengths.....	65
31. DEFVIEW return codes.....	66
32. DELACTQ operand value lengths.....	67
33. DELACTQ return codes.....	67
34. DELDSPS operand value lengths.....	68
35. DELDSPS return codes.....	69
36. DELIPCS operand value lengths.....	69
37. DELSRVM operand value lengths.....	70
38. DELSRVM return codes.....	71
39. DELVIEW operand value lengths.....	72
40. DELVIEW return codes.....	73
41. QUIT return codes.....	73
42. REVOKE operand value lengths.....	74
43. REVOKE return codes.....	75
44. STATUS operand value lengths.....	77
45. STATUS return codes.....	80
46. STOP return codes.....	80
47. CONFIG operand value lengths.....	82
48. CONFIG return codes.....	83

49. DEFACTN operand value lengths.....	88
50. DEFACTN return codes.....	89
51. DEFDATE operand value lengths.....	90
52. DEFDATE return codes.....	90
53. DEFEMON operand value lengths.....	93
54. DEFEMON return codes.....	94
55. DEFIMON operand value lengths.....	97
56. DEFIMON return codes.....	99
57. DEFGROUP operand value lengths.....	100
58. DEFGROUP return codes.....	101
59. DEFMMON operand value lengths.....	103
60. DEFMMON return codes.....	105
61. DEFPMON operand value lengths.....	107
62. DEFPMON return codes.....	109
63. DEFULE operand value lengths.....	112
64. DEFRULE return codes.....	115
65. DEFSCHD operand value lengths.....	118
66. DEFSCHD return codes.....	120
67. DEFSMON operand value lengths.....	122
68. DEFSMON return codes.....	123
69. DELACTN operand value lengths.....	124
70. DELACTN return codes.....	124
71. DELDATE operand value lengths.....	125
72. DELDATE return codes.....	126
73. DELEMON operand value lengths.....	126

74. DELEMON return codes.....	127
75. DELGROUP operand value lengths.....	127
76. DELGROUP return codes.....	128
77. DELIMON operand value lengths.....	129
78. DELIMON return codes.....	129
79. DELMMON operand value lengths.....	130
80. DELMMON return codes.....	130
81. DELPMON operand value lengths.....	131
82. DELPMON return codes.....	131
83. DELRULE operand value lengths.....	132
84. DELRULE return codes.....	132
85. DELSCHD operand value lengths.....	133
86. DELSCHD return codes.....	133
87. DELSMON operand value lengths.....	134
88. DELSMON return codes.....	134
89. RESUME operand value lengths.....	135
90. RESUME return codes.....	136
91. RUN operand value lengths.....	138
92. RUN return codes.....	139
93. STATUS operand value lengths.....	140
94. STATUS return codes.....	143
95. SUSPEND operand value lengths.....	144
96. SUSPEND return codes.....	145
97. Reserved data space names.....	147
98. ALTRCON operand value lengths.....	149

99. ALTRCON return codes.....	150
100. LOGTEXT operand value lengths.....	152
101. LOGTEXT return codes.....	153
102. VIEWCON operand value lengths.....	155
103. VIEWCON return codes.....	159
104. VIEWLOG operand value lengths.....	160
105. VIEWLOG return codes.....	165
106. VIEWSPL operand value lengths.....	166
107. Spool file list indicators.....	172
108. VIEWSPL return codes.....	176
109. Substitution variables used by commands.....	183
110. DEFEMON substitution variables.....	183
111. DEFEMON event substitution variables &6 and higher.....	185
112. DEFIMON substitution variables.....	189
113. DEFMMON substitution variables.....	189
114. DEFPMON substitution variables.....	189
115. DEFSMON substitution variables.....	190
116. DEFRULE substitution variables.....	190
117. DEFSCHD and RUN substitution variables.....	191
118. Accounting record format.....	193
119. Monitoring record format.....	194
120. Sample translation table files.....	201
121. GOMRSIF return codes.....	204
122. GOMGLBL return codes.....	211
123. Severity codes.....	215





## About this information

---

This book provides instructions for using IBM Operations Manager for z/VM. It is designed to help administrators and operators perform these tasks:

- Use commands to perform operations tasks, such as defining automation tasks, defining events that require automated responses, monitoring users' consoles, and automating console or terminal sessions.
- Configure Operations Manager to suit your environment.
- Use Operations Manager to monitor and manage z/VM systems and virtual machines.

Always check the VM Tools Library page for the most current version of this publication:

<https://www.ibm.com/us-en/marketplace/operations-manager-for-zvm>

## Service updates and Support information

---

To find service updates and support information, including software FixPaks, PTFs, Frequently Asked Question (FAQs), technical notes, troubleshooting information, documentation, and downloads, refer to the following Web page:

<http://www.ibm.com/software/products/en/operations-manager-for-zvm>



# Summary of Changes

---

This section summarizes the significant improvements or enhancements for IBM® Operations Manager for z/VM® V1.6 and refers you to relevant sections of this book for more information. Minor modifications to the text are not listed.

**Note:** Changes to V1.5 of this product can be found in previous releases of this guide.

## September, 2022 (PTF UI81961)

Support masking messages in the Operations Manager log.

- Added a masking character for messages in the Operations Manager log. For more information, see [DEFOPTN](#).
- Added the MSK input option as a well-known action. For more information, see [DEFACTN](#).
- Added the MASKNEXT *lines* operand to mask lines in the Operations Manager log. For more information, see [DEFRULE](#).

## March, 2022 (PTF UI79652)

New CHKDUMP SAMPEXEC. For more information, see [CHKDUMP SAMPEXEC](#).

## November, 2021 (PTF UI77841)

- Added new options to the VIEWCON command to support saving the current console to a file on disk. For more information, see [“VIEWCON” on page 153](#).
- Added new options to the VIEWLOG command to support saving the current Operations Manager log to a file on disk. For more information, see [“VIEWLOG” on page 159](#).
- Added new error message GOM0312E. For more information, see [“GOM0312E” on page 233](#).

## July, 2021 (PTF UI75725)

- New CHKSFS SAMPEXEC. For more information, see [CHKSFS SAMPEXEC](#).
- The VIEWSPL command is updated to display system data files in addition to regular spool files. VIEWSPL is also updated to display which volume(s) a spool file occupies and to allow selection of files having data allocated on particular volume(s).
  - A new operand (SVOL) has been added to the VIEWSPL command. Valid operand length of the SVOL operand is 1-55.
  - A new function has been added to toggle display format (PF04). This is active only on the spool list panel.
  - A new configuration statement (FORMAT=) is added to the VIEWSPL PROFILE.

For more information, see [“VIEWSPL” on page 165](#).

- The DEFOPTN command is updated to add a configuration option to control whether the system data files can be manipulated from the VIEWSPL spool file screen.
  - A new operand (SDFCMD) has been added to the DEFOPTN command. Valid operand length of the SDFCMD operand is 1-1.

For more information, see [“DEFOPTN” on page 57](#).

## December, 2020

A new sample EXEC has been added, GETSPCON EXEC, that allows you to capture spooled console data and add it to the corresponding service machine's console data, making it visible in the log and viewable with the VIEWCON and VIEWLOG commands. For more information, see [“GETSPCON SAMPEXEC” on page 178](#).

## July, 2020 (Documentation change only)

Numerous examples have been added to the documentation to help you understand what type of results you may see when issuing a command. For more information, see the "Example" section under any command.

## May, 2020 (PTF UI68310)

The following changes have been made to the documentation for this release:

- *Support encryption of TCP/IP connections.* Operations Manager now supports using SSL/TLS session encryption. TCP/IP sessions between Operations Manager instances, and TCP/IP sessions initiated by the general writer can be defined as secure connections. The DEFIPCS statements for these sessions (APPL=GOMRSCC, APPL=GOMRSCS, and APPL=GOMRSGW) now allow parameters to define them as secure, and to specify the label of the certificate to provide to the session partner. See the [DEFIPCS](#) statement for details.

**Note:** The parameter validation of all DEFIPCS statements was modified to thoroughly validate the content of the PARM operand. If an error is detected with the PARM content, message GOM0503E is displayed. Also messages GOM0411E and GOM0412E have been replaced by enhanced messages GOM0501E and GOM0502E respectively.

- *Pass last invoking information when a limit is reached.* When the LIMIT for a rule, schedule, or monitor is reached, the message field passed to the limit action now includes the message field passed to the previous processing ACTION. The previous message field is passed as "&6-&N" to the LIMIT action. See the DEFRULE, DEFSCHD, DEFEMON, DEFMMON, DEFPMON, DEFMON, and DEFIMON statements in [Chapter 8, “Syntax for Configuration Commands,” on page 81](#) for details.
- *VIEWCON interaction with z/OS guest.* VIEWCON may now be configured to send commands to a monitored guest using the simulated system console interface. This support is intended to allow Operations Manager to interact with any guest, such as z/OS, that supports receiving commands through the simulated system console.

To use this support, you must configure Operations Manager. Each monitored guest that is to receive commands on the simulated system console should be a member of the group distinguishing how VIEWCON sends commands to guests. The DEFOPTN command now includes the operand VCSCGRP, which specifies the name of a group containing the user IDs of the monitored guests that support receiving commands on the simulated system console. See the [DEFOPTN](#) command for details.

- *VIEWLOG and VIEWCON usage of XEDIT to inspect current content.* You may now pass the current view contents to an XEDIT session. When the XEDIT session completes, control is returned to the live screen. See the [VIEWLOG](#) and [VIEWCON](#) commands for details.
- *Messages.* A few messages have been added or changed for this release: GOM309E, GOM0310E, GOM0311E, GOM0501E, GOM0502E, GOM0503E, and GOM2887I.

## March, 2020 (PTF UI67484)

The following changes have been made to the documentation for this release:

- *Improved External Security Manager (ESM) granularity.* Typically, the ESM command authorization is used in combination with the RESUME and SUSPEND commands on any resource, such as a rule, schedule, or monitor. However, now authorization checking can be changed from the use of RESUME and SUSPEND commands to authorization checking at the resource type and resource name level. This is an optional feature for people who want to more concisely control the use of the RESUME and SUSPEND commands.

Use the DEFOPTN command to enable these changes, which are only applicable for ESM authorization. Options ESM Y, AUTHCMD Y, and AUTHEXT 1 must be active to enable these changes. Refer to [“DEFOPTN” on page 57](#) and to [“ESM class and Facility class profiles” on page 11](#) for more information.

**Note:** The DEFOPTN command with the ESMCLASS operand is available to configure the ESM class containing the profiles. The default DEFOPTN ESMCLASS is “FACILITY”. You may change the DEFOPTN ESMCLASS value if the default ESM class imposes restrictions on the profile naming length. For example, to use ESM class XFACILIT to contain the profiles, specify DEFOPTN ESMCLASS XFACILIT to Operations Manager and move all ESM profiles required by Operations Manager from the FACILITY class to the XFACILIT class in the ESM.

- *Improved action processing.* In the background processing, Operations Manager uses a TERM PRECEDENCE program to capture action output. The output from an action executed on an action processing machine is now captured locally, instead of being sent to the SECUSER or OBSERVER. The TERM PRECEDENCE support is only available in z/VM 7.1 and above, with PTFs UM35353 and UM35571. If TERM PRECEDENCE support is not available, this change has no effect.

**Note:** If there is no SECUSER or OBSERVER established for the action processing machine, this change will have no effect. If there is a SECUSER or OBSERVER established, this change may require configuration changes to Operations Manager.

If the action processing machine's SECUSER or OBSERVER target is an instance of Operations Manager, you should review any actions where the output triggers a rule. Any action which requires rule processing of the output should change the action definition to include the “OUTPUT ANL” operand. Adding “OUTPUT ANL” to the action will process the action output through the rules. See [“DEFACTN” on page 84](#) for more information on how to use the OUTPUT ANL option.

If the action processing machine has a SECUSER or OBSERVER directed to an Operations Manager instance, the action output is logged with message GOM0270I instead of message GOM0216L.

## September, 2019 (PTF UI65400)

The following changes have been made to the documentation for this release:

- *Added “Group” to some commands.* A new operand, GROUP *groupname*, has been added to the DEFVIEW, DELVIEW, DEFDSPS, and DELDSPS commands. Refer to [“DEFVIEW” on page 64](#), [“DELVIEW” on page 71](#), [“DEFDSPS” on page 50](#), and [“DELDSPS” on page 68](#) for more information.
- *Changes to Messages.* Four new messages have been added: GOM2890I, GOM2891I, GOM2892I, and GOM2893I. Two messages have been removed from the product: GOM0963I and GOM0964I.

## May, 2019 (PTF UI63172)

The following changes have been made to the documentation for this release:

- *Changes to Appendix C: Using substitution variables in commands.* A few changes and additions have been made to the tables in Appendix C to show new values for variables. See [Appendix C, “Using substitution variables in commands,” on page 183](#) for more information.
- *Correction to syntax.* The Name parameter in the CMDIPCS, DEFIPCS, and DELIPCS commands can be shortened to “NA” (NAme). The syntax diagrams have been changed to show the shortened version of the Name parameter. Refer to [“CMDIPCS” on page 150](#), [“DEFIPCS” on page 52](#), and [“DELIPCS” on page 69](#) for more information.
- *Improvement to VIEWSPL.* When a sort column is selected, the column heading is highlighted. See the VIEWSPL [“Usage notes” on page 171](#) for more information.
- *New messages.* A couple of new messages, GOM0271E and GOM0276E, have been added to the documentation. See [Appendix K, “Messages and Codes,” on page 215](#) for more information.

## November, 2018

The following changes have been made to the documentation for this release:

- Additional information has been supplied about the substitution variables used by the DEFEMON command. See [Appendix C, “Using substitution variables in commands,”](#) on page 183 for more information.
- Additional information has been provided to explain why certain privilege classes are required for various operands. See [“Updates to CP directory”](#) on page 6 for more information.
- Additional information has been provided to explain when to use two single quotes ( ' ') as part of command syntax. See [“Command syntax rules”](#) on page 31 for more information.
- Additional information has been provided about the CMDIPCS command. See [“CMDIPCS”](#) on page 150 for more information.
- Several of the examples have been revised to more clearly show when "VMLINK" is used as part of the command.

## December, 2017

The following changes have been made to the documentation for this release:

- For the DEFACTN and RUN commands, the COMMAND operand has been changed to allow a prefix of "CP". This will cause the COMMAND operand to be processed through the CP diagnose x'0008' interface to send case-sensitive data to z/VM guests, such as Linux. See [“Usage notes”](#) on page 88 for more information.
- Additional information has been added to the "Usage Notes" for the VIEWSP command to clarify how an asterisk (\*) is treated. See [“Usage notes”](#) on page 171.
- A few new messages were added. See [Appendix K, “Messages and Codes,”](#) on page 215 for more information.

## January, 2017

The following changes have been made to the documentation for this release:

- Several new operands have been added to the DEFOPTN command to specify the installation defaults for limits. The new operands include EMONLIM, IMONLIM, MMONLIM, PMONLIM, SMONLIM, RULELIM, and SCHDLIM. These limits can also be modified by the LIMIT operand in the DEFEMON, DEFIMON, DEFMMON, DEFPMON, DEFMON, DEFRULE, and DEFSCHD commands. See the [“DEFOPTN”](#) on page 57 for more information on the new operands.
- The LIMIT operand has been changed to include an optional limit action, *lmtactn*. This action is processed once if the limit condition is encountered within a LIMIT interval. The *lmtactn* parameter is included in the DEFEMON, DEFIMON, DEFMMON, DEFOPTN, DEFPMON, DEFRULE, DEFSCHD, DEFMON commands. Refer to any of these commands for more information on the LIMIT operand.

## October, 2016

The following changes have been made to the documentation for this release:

- A new action, NULL, has been added to several of the commands, including DEFACTN, DEFEMON, DEFIMON, DEFMMON, DEFPMON, DEFRULE, DEFSCHD, DEFMON, DELACTN, and RUN. See [“DEFACTN”](#) on page 84 for more information about NULL.
- The DEFIPCS command has changed to support receiving syslog data using TCP protocol. Refer to [“DEFIPCS”](#) on page 52 and to [“Collection and processing of syslog data”](#) on page 200 for more information. The PTF for APAR PI74064 is required to use this function.
- A new return code, 103, has been added to the GOMCMD, GOMMAIN, and GOMSVM commands. This return code indicates that the machine architecture of the virtual machine is not compatible with Operations Manager.
- A few messages have been added or changed for this release: GOM0008E, GOM0014E, GOM0274W and GOM0275W. Refer to [Appendix K, “Messages and Codes,”](#) on page 215 to see the messages.

## May, 2016

For the initial release of Version 1.6, every command listed in this book has been reviewed and revised. Numerous messages have also been improved for clarity. Additional changes are listed below:

- A new command, DELATCQ, has been added. This command deletes an entry from the processing list. For more information, see [“DELACTION” on page 66](#).
- CMDTCPA, DEFTCPA, and DELTCPA are no longer supported. These commands have been replaced with CMDIPCS, DEFIPCS, and DELIPCS. For information on these new commands, see [“CMDIPCS” on page 150](#), [“DEFIPCS” on page 52](#), and [“DELIPCS” on page 69](#).
- A new appendix describes which variables may be substituted with a value when an action is triggered by a rule, schedule, and monitor. For details, see [Appendix C, “Using substitution variables in commands,” on page 183](#).
- A new appendix explains how wildcard pattern matching is performed, such as which symbols can be used as wildcard characters. For more information, see [Appendix B, “Wildcard pattern matching,” on page 181](#).
- The new syntax style for commands no longer requires commas or parentheses, with very few exceptions, making it easier to type commands. For compatibility, Operations Manager V1.6 also accepts commands in the syntax style of earlier releases, which uses parentheses as operand value delimiters and commas as operand keyword separators. For additional information on command syntax, see [“Command syntax rules” on page 31](#).
- Additional information has been provided on how to resolve domain system names (DSNs) and share action processing environment for multiple instances. For details, see [Appendix F, “Using internet protocol and networked components,” on page 197](#).
- Several new messages have been added and many changes were made to existing messages. For information on any message that is generated by Operations Manager, see [Appendix K, “Messages and Codes,” on page 215](#).
- The USER operand is no longer supported on the DEFGROUP and DELGROUP commands. Use the MEMBER operand instead. For details, see [“DEFGROUP” on page 99](#) and [“DELGROUP” on page 127](#).
- The structure and operands for defining user authorization using the AUTH and REVOKE commands are not backward compatible. Refer to [“AUTH” on page 43](#) and [“REVOKE” on page 74](#) for information on the new structure and syntax.
- The substitution variable "&S" is no longer supported. Use &SYSNAME instead. Refer to [Appendix C, “Using substitution variables in commands,” on page 183](#) for more information.
- On the DEFACTN statement, the INPUT and OUTPUT operands no longer support obsolete values. This includes the LOG option on INPUT and colors, audio, highlighting, holding, logging, and suppression on OUTPUT. Refer to [“DEFACTN” on page 84](#) for more information.
- The substitution variables available for actions associated with DEFPMON and DEFMON have changed.

In Operations Manager V1.5.0:

- DEFPMON - &6 was unassigned.
- DEFMON - &8 was unassigned.

In Operations Manager V1.6.0:

- DEFPMON - &6 contains the interval of the page space monitor.
- DEFMON - &8 contains the interval of the spool space monitor.

For more information on these commands, refer to [“DEFPMON” on page 105](#) and [“DEFMON” on page 120](#).





---

# Chapter 1. Introduction to Operations Manager

This chapter introduces you to the functionality of IBM Operations Manager for z/VM, including its major components and some key benefits of using it. For easier reading, the product name will be shown as Operations Manager throughout the documentation.

## **Related concepts**

[What does Operations Manager do?](#)

[Benefits of using Operations Manager](#)

## **Related reference**

[Operations Manager components](#)

[Prerequisites](#)

---

## What does Operations Manager do?

Operations Manager is designed to improve the monitoring and management of z/VM systems and virtual machines. This is accomplished by automating routine maintenance tasks, sending alerts based on situations or thresholds, and automatically responding to situations requiring intervention.

Operations Manager provides a set of commands that allow you to define automation tasks, define events that require automated responses, monitor users' consoles, and automate console or terminal sessions.

Operations Manager provides these functions:

- *Automating Tasks.* Commands are provided to define schedules and actions. *Schedules* define when to issue an *action* containing routine commands. An action can be issued once, on a specific hour or hourly, on a specific day or daily, on a holiday, monthly, or yearly.
- *Automating Responses.* Commands are provided to define rules and actions. *Rules* identify which messages should be acted upon and define a pattern to match against the messages. When a pattern match is detected, an action is scheduled to issue the appropriate commands.
- *Monitoring.* Commands are provided to monitor a user's console, spool usage, page space, and events. Multiple administrators may interact with a user's console simultaneously as if they were logged onto the user's account.
- *Automating sessions.* Commands are provided to control sessions that will perform tasks at specific times or off-shift without manual intervention. For example, Operations Manager can automatically log on to a user ID, perform a set of routine tasks, log off, and then terminate the session.

---

## Benefits of using Operations Manager

Operations Manager provides several benefits to z/VM operators and administrators:

- *Improved operations accuracy.* By automating routine tasks, Operations Manager removes the responsibility for administrators to issue routine commands repetitively, correctly, and at the appropriate time.
- *Reduced operations requirements.* By automating the response to situations that require intervention, Operations Manager eliminates the requirement for operators to correctly interpret messages, determine corrective actions, and correctly respond to situations.
- *Improved administrative notification.* When a situation occurs that requires intervention, Operations Manager can interact with a variety of external methods to notify key personnel about an event. This allows real-time notification about situations occurring in critical services (such as networking) or critical components (such as a spool) that require expert attention.

## Operations Manager components

---

Operations Manager consists of four separate components. These components work together to do all of Operations Manager message and action processing. Each of these components is described below.

### **Related reference**

[Application and settings interference](#)

## Message processing component

The GOMMAIN module, which runs on the OPMGRM1 user ID, acts as a central message processing service. It activates the framework required to receive console messages, process rules and messages, monitor user IDs, monitor spool and page space, run schedules, and run the actions associated with them.

In order to start the GOMMAIN module, you must issue the GOMMAIN command from the command line or an EXEC on the OPMGRM1 user ID. The GOMMAIN command references a start up configuration file containing various commands Operations Manager will use for its monitoring and message processing. Operations Manager provides the sample PROFILE EXEC for OPMGRM1 and a sample configuration file OPMGRM1 CONFSAMP. In addition, a second configuration file named EXAMPLES CONFSAMP contains examples of rules, monitors, and actions.

### **Related reference**

[GOMMAIN](#)

## Action processing component

The GOMSVM module, which runs on action processing servers (OPMGRS1, OPMGRS2, etc.), processes the actions you defined with various Operations Manager commands.

### **Related reference**

[GOMSVM](#)

## User interaction component

The GOMCMD module provides a way for users to interact with Operations Manager without starting or stopping it. Users use the GOMCMD command to dynamically issue any Operations Manager control or configuration commands from a user's console.

### **Related reference**

[GOMCMD](#)

## Automated session component

The GOMTDM module provides a way to automatically activate a session in response to a situation (message). For example, administrators can have Operations Manager log on to a user ID, perform a set of routine tasks, log off, and then disconnect.

### **Related reference**

[GOMTDM](#)

## Component interaction

The components of Operations Manager communicate with each other using Inter-User Communications Vehicle (IUCV). The following diagram shows their relationship and interaction:

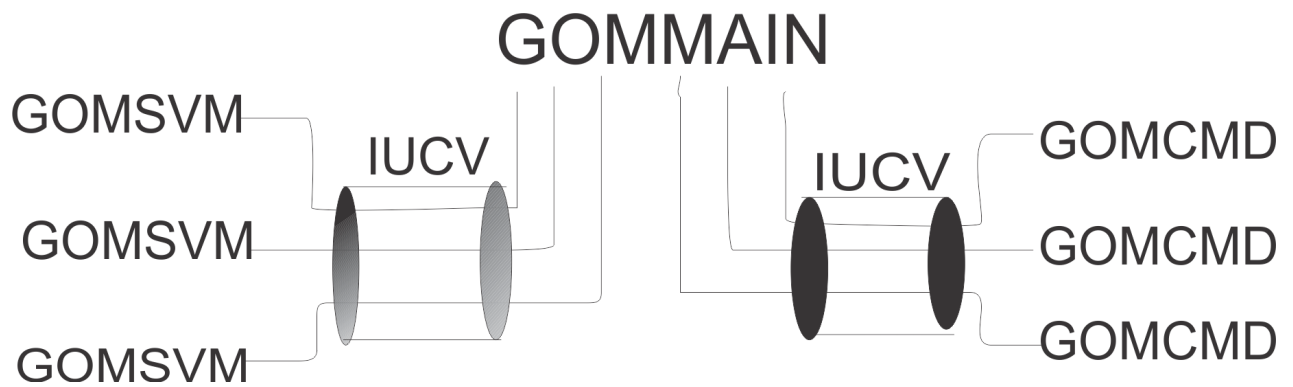


Figure 1. Operations Manager component interaction

## Accessing Help

---

Access Operations Manager help in the following ways:

- To access the help menu, type:

```
HELP AGOM
```

- To access help for a specific command, type "HELP AGOM" followed by the command name. For example:

```
HELP AGOM AUTH
```

## Prerequisites

---

### Hardware requirements

Operations Manager runs on any hardware that supports the required version of z/VM.

### Software requirements

Operations Manager runs on all currently supported versions and releases of z/VM.



---

## Chapter 2. Configuring Operations Manager

To install Operations Manager, follow the installation instructions provided in the *Operations Manager for z/VM Program Directory (GI10-8664)*. After you install Operations Manager, perform the steps described in this chapter to configure Operations Manager for your environment.

### Related tasks

[Step 1. Update OPMGRM1](#)

[Step 2. Update Action Processing Servers](#)

[Step 3. Update End User and Administrator IDs](#)

[Step 4. Update IDs monitored by Operations Manager](#)

[Step 5. \(Optional\) Configure an external security mechanism](#)

---

## Step 1. Update OPMGRM1

### Updates to PROFILE EXEC

You must create a PROFILE EXEC for any user ID that will be running GOMMAIN. The default user ID is OPMGRM1.

To create the PROFILE EXEC on a system with Operations Manager installed on minidisks, follow these steps:

1. Logon to MAINT or MAINT $vr$ m, where  $v$  is version,  $r$  is release, and  $m$  is modification, such as MAINT710 for version 7, release 1, modification 0.
2. VMLINK 5697J10F 2C2 < \* E RR >
3. VMLINK OPMGRM1 191 < \* F MR >
4. COPYFILE PROFM1 SAMPEXEC E PROFILE EXEC F
5. VMLINK 5697J10F 2C2 < DET >
6. VMLINK OPMGRM1 191 < DET > ( WRITE

To create the PROFILE EXEC on a system with Operations Manager installed in SFS directories, replace steps 2, 3, 5, and 6 with the appropriate VMLINK commands for the SFS directories.

The PROFILE EXEC for OPMGRM1 and any other user ID that will be running GOMMAIN should include the following statements:

```
/* IBM Operations Manager for z/VM (5697-J10) */
/* Sample lines to include in OPMGRM1 PROFILE EXEC */
Trace 0
Address Command
'CP SET RUN ON'
'ACCESS 194 D'
'ACCESS 198 E'
'ACCESS 400 G'
'SET LANGUAGE ( ADD GOM USER )'

/* Replace filename, filetype, fm with startup configuration file */
/* Replace the default logsize option of 4M with appropriate value */
'GOMMAIN OPMGRM1 CONFIG E ( 4M'

/* If the virtual console is not connected, logoff */
coninfo = Diagrc('24','1')
concond = Substr(coninfo,11,1)
If concond <> 0 Then
'CP LOGOFF'
Exit
```

For the GOMMAIN statement, replace OPMGRM1 CONFIG E with the file name, file type, and file mode of the startup configuration file. We recommend using OPMGRM1 CONFIG E or &SYSNAME CONFIG E where &SYSNAME will be resolved by Operations Manager to your system name (node ID). Refer to [“Copying the sample configuration file” on page 21](#) for more information on the configuration file name.

These sample statements can be found in the file PROFM1 SAMPEXEC provided on the Operations Manager samples disk 5697J10F 2C2, or the associated SFS directory.

### Related reference

[GOMMAIN](#)

[Creating a startup configuration file](#)

## Updates to CP directory

OPMGRM1, and any other user ID running GOMMAIN, requires the following CP directory statements. These sample statements are located in the file OPMGRM1 SAMPDIR provided on the Operations Manager samples disk 5697J10F 2C2 or associated SFS directory.

```
IDENTITY OPMGRM1 XXXXXXXX 64M 1024M ABCDG
  BUILD ON * USING SUBCONFIG OPMGM1-1
  ACCOUNT 5697-J10 GOM
  MACHINE XC
  XCONFIG ACCESSLIST ALSIZE 1022
  XCONFIG ADDRSPACE MAXNUMBER 1022 TOTSIZE 4G SHARE
  IUCV ALLOW MSGLIMIT 65535
  IUCV ANY MSGLIMIT 65535
  IUCV *MSG MSGLIMIT 65535
  IUCV *SPL MSGLIMIT 65535
  IUCV *VMEVENT MSGLIMIT 65535
  OPTION ACCT APPLMON
  IPL CMS
  CONSOLE 0009 3215
  SPOOL 000C 2540 READER *
  SPOOL 000D 2540 PUNCH A
  SPOOL 000E 1403 A
  LINK MAINT 190 190 RR
  LINK MAINT 19E 19E RR
  LINK MAINT 19D 19D RR
  MDISK 198 3390 <startcyl> 0005 <volume> RR <readpw> <writepw> <multpw>
SUBCONFIG OPMGM1-1
  MDISK 191 3390 <startcyl> 0005 <volume> MR <readpw> <writepw> <multpw>
  MDISK 194 3390 <startcyl> 0050 <volume> MR <readpw> <writepw> <multpw>
  MDISK 400 3390 <startcyl> 0005 <volume> RR <readpw> <writepw> <multpw>
  MDISK 410 3390 <startcyl> 0005 <volume> RR <readpw> <writepw> <multpw>
```

The CP directory for the user ID OPMGRM1, and any other user ID running GOMMAIN, requires OPTION ACCT to allow it to use DIAG x '4C' and generate accounting records. This statement is only required if you want accounting records to be created.

The CP directory for the user ID OPMGRM1, and any other user ID running GOMMAIN, requires OPTION APPLMON to allow it to use DIAG x 'DC' and generate monitor records. This statement is only required if you want monitoring records to be created.

Privilege classes are required to use certain components of Operations Manager. The required classes are listed below.

- Privilege class A is required for the FORCE command. The Operations Manager command, DELSRVM, deletes an action processing service machine from the current Operations Manager configuration. When the CP FORCE Y operand is specified, this ensures that the deletion will occur.
- Privilege class A or B is required for the XAUTOLOG command. The Operations Manager command, DEFSRVM, issues CP XAUTOLOG if the DEFSRVM command specifies the XAUTOLOG Y option. OPMGRM1 uses XAUTOLOG to automatically start the action processing servers, according to the option specified in the DEFSRVM command.

- Privilege class A, B, or C is required for the QUERY SECUSER and QUERY OBSERVER commands. The Operations Manager command, VIEWCON, displays the current console content for a monitored user, and issues CP QUERY SECUSER and CP QUERY OBSERVER for the specified USER/USERS.
- Privilege class C is required for the SEND command. While a user is viewing a console (using VIEWCON), Operations Manager issues CP SEND to process any console commands entered on the command line.
- Privilege class D is required for the CHANGE, TRANSFER, and PURGE commands related to spool files. OPMGRM1 issues these commands in response to requests from users of the VIEWSPL command. The VIEWSPL command issues CP CHANGE, CP TRANSFER, or CP PURGE for the file being processed.
- Privilege class D is required for the following QUERY commands:
  - ALLOC PAGE
  - MAXSPOOL
  - FILES
  - ALLOC SPOOL

The Operations Manager command, DEFPMON, issues CP QUERY ALLOC PAGE ALL to determine page space information. The DEFMON and VIEWSPL commands issue CP QUERY ALLOC SPOOL ALL, CP QUERY MAXSPOOL SYSTEM, and CP QUERY FILES SYSTEM ALL to determine spool space information.

- Privilege class B is required for the CP diagnostic code X'00D4'. The Operations Manager command, DEFACTN, is used to define an action to process as the result of a rule, schedule, or monitor. DEFACTN issues diagnose X'00D4' if the ALTUSER/SECLABEL options are specified.
- Privilege class D is required for the CP diagnostic code X'00D8'. The Operations Manager command, VIEWSPL, issues diagnose X'00D8' to determine spool file information.
- Privilege class B or E is required for the CP diagnostic code X'02CC'. The Operations Manager command, VIEWSPL, issues diagnose X'02CC' to determine spool file information.

The user ID running GOMMAIN also needs authority to run any actions using the ENV LVM operand. This operand specifies that the command is to be processed by the local virtual machine (OPMGRM1). Refer to “Operands” on page 85 for more information on the ENV LVM operand.

## Step 2. Update Action Processing Servers

---

### Updates to PROFILE EXEC

You must create a PROFILE EXEC for any user ID that will be running GOMSVN to process actions. The default user IDs are OPMGRS1 through OPMGRS4.

To create the PROFILE EXEC on a system with Operations Manager installed on minidisks, follow these steps:

1. Logon to MAINT or MAINT $_{vrm}$ , where  $v$  is version,  $r$  is release, and  $m$  is modification, such as MAINT710 for version 7, release 1, modification 0.
2. VMLINK 5697J10F 2C2 < \* E RR >
3. VMLINK OPMGRS1 191 < \* F MR >
4. COPYFILE PROFS1 SAMPEXEC E PROFILE EXEC F
5. VMLINK OPMGRS1 191 < DET > ( WRITE
6. Repeat steps 3 - 5 for each action processing server.
7. VMLINK 5697J10F 2C2 < DET >

To create the PROFILE EXEC on a system with Operations Manager installed in SFS directories, replace steps 2, 3, 5, and 7 with the appropriate VMLINK commands for the SFS directories.

The PROFILE EXEC for OPMGRS1 and any other user ID that will be running GOMSVN should include the following statements:

```
/* IBM Operations Manager for z/VM (5697-J10) */
/* Sample lines to include in OPMGRS1-4 PROFILE EXEC */

Trace 0
Address Command
'CP SET RUN ON'
'ACCESS 198 E'
'ACCESS 400 G'
'SET LANGUAGE ( ADD GOM USER )'

'GOMSVN OPMGRM1'

/* If the virtual console is not connected, logoff */
coninfo = Diagrc('24','-1')
concond = Substr(coninfo,11,1)
If concond <> 0 Then
  'CP LOGOFF'
Exit
```

These sample statements can be found in the file PROFS1 SAMPEXEC provided on the Operations Manager samples disk 5697J10F 2C2, or the associated SFS directory.

**Note:** When OPMGRM1 (or any server running GOMMAIN) is stopped, any action processing servers connected to it are also stopped. As shown above, if the PROFILE EXEC on each action processing server contains 'CP LOGOFF', and the action processing server is running disconnected at the time, the action processing servers are automatically logged off when OPMGRM1 is stopped.

## Updates to CP directory

User IDs OPMGRS1–4 require the following CP directory statements. These sample statements are located in the file OPMGRS1 SAMPDIR provided on the Operations Manager samples disk 5697J10F 2C2 or associated SFS directory.

```
IDENTITY OPMGRS1 XXXXXXXX 32M 1024M G
BUILD ON * USING SUBCONFIG OPMGS1-1
ACCOUNT 5697-J10 GOM
MACHINE XC
XCONFIG ACCESSLIST ALSIZE 126
XCONFIG ADDRSPACE MAXNUMBER 126 TOTSIZE 512M SHARE
IUCV ALLOW MSGLIMIT 65535
IUCV ANY MSGLIMIT 65535
IUCV *MSG MSGLIMIT 65535
IPL CMS
CONSOLE 0009 3215 T OPMGRM1
SPOOL 000C 2540 READER *
SPOOL 000D 2540 PUNCH A
SPOOL 000E 1403 A
LINK MAINT 190 190 RR
LINK MAINT 19E 19E RR
LINK MAINT 19D 19D RR
LINK OPMGRM1 198 198 RR
LINK OPMGRM1 400 400 RR
SUBCONFIG OPMGS1-1
MDISK 191 3390 <startcyl> 0005 <volume> MR <readpw> <writepw> <multpw>
```

The user ID running GOMSVN also needs authority to run any actions using the ENV SVM operand. This operand specifies that the command is to be processed by any available action processing server. Refer to [“Operands” on page 85](#) for more information on the ENV SVM operand.

## Step 3. Update End User and Administrator IDs

**Note:** If the USER components of Operations Manager were copied to MAINT 19D/19E during product installation, you can omit the LINK and ACCESS statements from the sample below. However, you must include the SET LANGUAGE statement. For more information on the USER components and MAINT 19D/19E, see *"Copy Operations Manager Files Into Production,"* in the *Operations Manager for z/VM Program Directory (GI10-8664)*.



You must update the PROFILE EXEC for any user ID that will be using GOMCMD to issue commands or GOMTDM to automate console or terminal sessions. The PROFILE EXEC for these user IDs should include the following statements:

```
/* IBM Operations Manager for z/VM (5697-J10) */
/* Sample lines to include in user's PROFILE EXEC to use GOMCMD */
/* Sample lines to include in user's PROFILE EXEC to use GOMTDM */

'CP LINK OPMGRM1 410 410 RR'
'ACCESS 410 G'
'SET LANGUAGE ( ADD GOM USER )'
```

These sample statements can be found in the PROFUSR SAMPEXEC provided on the Operations Manager samples disk 5697J10F 2C2 or associated SFS directory.

**Note:** Privilege class G is required for the user ID running GOMCMD.

## Step 4. Update IDs monitored by Operations Manager

Each user ID console that you want Operations Manager to monitor must be updated. Set the SECUSER or OBSERVER user ID to OPMGRM1, or the name of the user ID running GOMMAIN. You must update either the CP directory or the PROFILE EXEC of each monitored user ID. IBM recommends updating the CP directory, allowing Operations Manager to see all messages on the console from the start of the logon process.

- If you use OBSERVER for monitoring users, message HCP150A is not generated. If you use SECUSER, message HCP150A is generated.
- If you use SECUSER, console messages are not sent to the target user ID unless the source user ID is running disconnected. If you use OBSERVER, console messages are sent to the target user ID when both the source user ID is running disconnected and when it is logged onto a console.
- If you use SECUSER, Operations Manager can send commands to the source user ID (through Operations Manager actions) without the need for an additional privilege class. If you use OBSERVER, Operations Manager servers must be able to issue the CP SEND command. By default this requires privilege class C.
- If a disconnected user has a SECUSER or OBSERVER defined and also runs an application which connects to the CP \*MSG service in order to capture CPCONIO and/or VMCONIO output to the virtual machine console, some or all of that output may be sent to the SECUSER or OBSERVER rather than the local \*MSG connection. This may result in the application operating incorrectly.
- Refer to Appendix H, “Application and settings interference,” on page 207 for more information on how OBSERVER and SECUSER settings can impact Operations Manager.

### Updating the CP directory

To update the CP DIRECTORY of the monitored user ID, you might use one of the following commands:

```
CONSOLE 009 3215 T OPMGRM1
```

or

```
CONSOLE 009 3215 T OPMGRM1 OBSERVER
```

For more information, see the CONSOLE statement in *z/VM CP Planning and Administration (SC24-6178)*.

### Updating the PROFILE EXEC

If you choose to update the PROFILE EXEC of the monitored user ID, instead of updating the CP directory entry, you might use one of the following commands:

```
CP SET SECUSER OPMGRM1
```

or

CP SET OBSERVER OPMGRM1

For more information, see the SET SECUSER command in the *z/VM CP Command and Utility Reference (SC24-6175)*.

## Verifying the console data is being sent to Operations Manager

To verify that the console data for a monitored user ID is being sent to Operations Manager, follow the steps below.

If you update the CP directory for this user ID to have the console data monitored by Operations Manager, it does not go into effect until the user ID logs off and logs on again (or re-IPLs). The SET SECUSER command causes the user ID to immediately start sending console data to Operations Manager. The next time that the user ID is restarted, the updated directory entry takes effect.

1. Log on to an authorized user ID. This user ID must be authorized to issue the CP QUERY SECUSER and CP SET SECUSER commands.
2. Select a user ID that you configured to have its console monitored by Operations Manager. Refer to [“Step 4. Update IDs monitored by Operations Manager”](#) on page 9 for more information.
3. Issue the following command to check if the user ID is being monitored by Operations Manager:

```
CP QUERY SECUSER userid
```

4. Follow the appropriate condition:

- If the user ID is already being monitored by Operations Manager, no action is needed.
- If the user ID is not being monitored by Operations Manager, issue the following command:

```
CP SET SECUSER userid OPMGRM1
```

## Step 5. (Optional) Configure an external security mechanism

If you use an ESM (external security manager) that supports the RACROUTE macro and the FACILITY class, you can configure Operations Manager to use the ESM as its authorization mechanism instead of using its own internal authorization mechanism. You can control which users have authority to issue which commands through the ESM, as well as which users can view the consoles and spool files of specific or all users.

## Set up example for IBM Resource Access Control Facility (RACF)

OPMGRM1, and any other user ID running GOMMAIN, must have authority to issue checks for access to a resource on behalf of another user. An example on how to do this for IBM Resource Access Control Facility (RACF) is provided, however, refer to the latest documentation for your particular release of ESM for details.

To use RACF as the authorization mechanism, perform the following steps:

1. Authorize communication with the RACF/VM server through IUVC.
2. Authorize third-party authorization checks.
3. Identify the RACF/VM server to which RACROUTE will be sent.
4. Define ESM options for Operations Manager.

All of these steps are described below.

### Authorize communication with RACF/VM

Servers communicate with RACF through IUCV. There are two options for authorizing this communication. You should consider which option is appropriate for your system's security administration policies.

- The more general option is to have the RACF/VM server (usually called RACFVM) include an IUCV ALLOW statement in its directory entry. This will permit any user in the system to establish communication with the RACF server.
- The more restricted option is to have the Operations Manager service machine, OPMGRM1, include an IUCV statement for the RACF/VM server (for example, IUCV RACFVM PRIORITY MSGLIMIT 255) in its directory entry. This will permit only the users with this statement in their directory entry to establish communication with the RACF server.

## Authorize third-party checks

Authority to issue third-party authorization checks is controlled by the ICHCONN profile in the FACILITY class in RACF/VM. You must create this profile if it does not exist and give the Operations Manager server UPDATE access to it. The ICHCONN profile must be in the FACILITY class.

## Identify the RACF/VM server

The RACROUTE interface must be able to identify the RACF/VM server. This is done with the RACF SERVMACH file, which is normally loaded to the CMS Y-disk during RACF installation. If you want to use a different RACF server, or your installation has removed the RACF SERVMACH file from general access, you should provide a tailored RACF SERVMACH file to Operations Manager's main server OPMGRM1.

## Define ESM options for Operations Manager

To activate the use of an external security manager, you must set the ESM operand value to Y on the DEFOPTN command in the Operations Manager startup configuration file.

```
DEFOPTN ESM Y
```

Even with ESM set to Y, it is recommended that at least one user be given access to Operations Manager in case ESM must be set to "N" temporarily. Refer to [“Authority assignments with AUTH and external security”](#) on page 24 for more information.

Refer to [DEFOPTN Operands](#) for more information on the available operands.

## Set the ESMCLASS option

If you set DEFOPTN ESM to Y, you can use the ESMCLASS operand to configure the ESM class containing the Operations Manager profiles. By default, the FACILITY class is used to contain the Operations Manager ESM profiles. It is highly recommended to use the default FACILITY class, unless you experience problems that require an alternate ESM class.

## Set the ESMPREF option

If you set DEFOPTN ESM to Y, you can use the ESMPREF operand to configure the prefix Operations Manager adds to the RACF profile names. Operations Manager automatically adds "GOM." to the front of your prefix. For example, if you specify ESMPREF OPSMGR, the ESMCLASS profiles must be defined with the prefix GOM.OPSMGR.

If you set DEFOPTN ESM to Y and do not configure the ESMPREF operand, the default prefix is RKT.OPM. In this case, the ESMCLASS profiles must be defined with the prefix RKT.OPM

Refer to [DEFOPTN Operands](#) for more information on using the ESMPREF operand.

## ESM class and Facility class profiles

All of the resources Operations Manager uses are profiles in the ESM class as defined by the DEFOPTN ESMCLASS operand. These profiles determine users' level of authorization.

Replace ESMPREF in the tables below with one of the following:

- RKT.OPM – This is the default. Use this value if you have not specified a value for ESMPREF on the DEFOPTN statement in the Operations Manager configuration file.
- GOM.*prefix* – where *prefix* is a name of your choosing. The *prefix* must match the value specified for ESMPREF on the DEFOPTN statement in the Operations Manager configuration file.

Replace ESMCLASS in the tables below with one of the following:

- FACILITY – This is the default. Use this value if you have not specified a value for EMSCLASS on the DEFOPTN statement in the Operations Manager configuration file.
- *esmclass* – where *esmclass* is the name of the ESM class containing Operations Manager profiles.

Operations Manager can be configured to check profiles at a general level or by specific commands. The following sections describe the available profiles for each option.

## General profiles

Operations Manager checks these profiles when **all** of the following are true:

- DEFOPTN ESM is set to Y.
- AUTHCMD is set to N.
- AUTHEXT is set to 0 or is not used.

Table 1. General ESMCLASS profiles	
Profile	Authorization
ESMPREF.CONTROL	Users who have READ access to this profile have authority to issue control and configuration commands.
ESMPREF.CONFIG	Users who have READ access to this profile have authority to issue configuration commands.
ESMPREF.ALTRCON. <i>userid</i> Refer to the note below for the valid replacements of <i>userid</i> .	Users who have READ access to this profile have authority to alter the display attributes of a line in a user's console when it is displayed using VIEWCON.
ESMPREF.VIEWCON. <i>userid</i> Refer to the note below for the valid replacements of <i>userid</i> .	Users who have READ access to this profile have authority to view consoles.
	Users who have UPDATE access to this profile have authority to view consoles and issue commands.
ESMPREF.VIEWSPL. <i>userid</i> Refer to the note below for the valid replacements of <i>userid</i> .	Users who have READ access to this profile have authority to view spool files.
	Users who have UPDATE access to this profile have authority to view spool files, purge spool files, and alter a spool file's external attributes.
ESMPREF.CMDIPCS. <i>ipcsid</i> Refer to the note below for the valid replacements of <i>ipcsid</i> .	Users who have READ access to this profile have authority to interact with the IP session defined as <i>ipcsid</i> .

The *userid* value in the *ESMPREF.ALTRCON.userid*, *ESMPREF.VIEWCON.userid*, and *ESMPREF.VIEWSPL.userid* profiles can be replaced with a wildcard pattern or one of the following options:

- Any individual user ID. For example, READ access to profile *ESMPREF.VIEWCON.JONES* allows the user to view the console of the user ID JONES.

- Any combined view defined by the DEFVIEW command. For example, READ access to profile *ESMPREF.VIEWCON.VIEW3* allows the user to view the combined view named VIEW3, which contains USER1, USER2, and USER3.
- A generic matching pattern. For example, READ access to profile *ESMPREF.VIEWCON.\** allows the user to view the console of any user monitored by Operations Manager. Before defining any generic profiles in the ESMCLASS, ensure that the ESMCLASS can contain generic profiles. Refer to *RACF® Security Server Security Administrator's Guide (SC24–6142)*.

The *ipcsid* value in the *ESMPREF.CMDIPCS.ipcsid* can be replaced with a wildcard pattern or one of the following options:

- Any individual IP session name. The IP session name is defined by the DEFIPCS command NAME operand. For example, READ access to profile *ESMPREF.CMDIPCS.SYS23* allows the user to interact with the IP session defined with DEFIPCS NAME SYS23.
- A generic matching pattern. For example, READ access to profile *ESMPREF.CMDIPCS.\** allows the user to interact with any IP session defined by DEFIPCS. Before defining any generic profiles in the ESMCLASS, ensure that the ESMCLASS can contain generic profiles. Refer to *RACF Security Server Security Administrator's Guide (SC24–6142)*.

## Command-level profiles

By setting DEFOPTN ESM to Y and AUTHCMD to Y, Operations Manager provides the ability to authorize by specific commands.

### Notes:

1. If DEFOPTN ESM is set to Y and AUTHCMD is set to Y, Operations Manager does *not* check these profiles:
  - *ESMPREF.CONTROL*
  - *ESMPREF.CONFIG*
2. Generic profiles can also be used based on your External Security Manager. For example, READ access to profile *ESMPREF.COMMAND.\** gives a user the authority for all Operations Manager control and configuration commands. Before defining any generic profiles in the ESMCLASS, ensure that the ESMCLASS can contain generic profiles. Refer to *RACF Security Server Security Administrator's Guide (SC24–6142)*.

If DEFOPTN ESM is set to Y and AUTHCMD is set to Y, Operations Manager checks the profiles shown in the table below.

Table 2. Command-level ESMCLASS profiles	
Profile	Authority
<i>ESMPREF.VIEWCON.userid</i> Refer to the note above for the valid replacements of <i>userid</i> .	Users who have READ access to this profile have authority to view consoles.
	Users who have UPDATE access to this profile have authority to view consoles and issue commands.
<i>ESMPREF.ALTRCON.userid</i> Refer to the note above for the valid replacements of <i>userid</i> .	Users who have READ access to this profile have authority to alter the display attributes of a line in a user's console when it is displayed using VIEWCON.
<i>ESMPREF.VIEWSPL.userid</i> Refer to the note above for the valid replacements of <i>userid</i> .	Users who have READ access to this profile have authority to view spool files.
	Users who have UPDATE access to this profile have authority to view spool files, purge spool files, and alter a spool file's external attributes.

Table 2. Command-level ESMCLASS profiles (continued)

Profile	Authority
ESMPREF.CMDIPCS.ipcsid Refer to the note above for the valid replacements of ipcsid.	Users who have READ access to this profile have authority to interact with the IP session defined as ipcsid.
ESMPREF.COMMAND.AUTH	Users who have READ access to this profile have authority for the AUTH command.
ESMPREF.COMMAND.CMS	Users who have READ access to this profile have authority for the CMS command.
ESMPREF.COMMAND.CONFIG	Users who have READ access to this profile have authority for the CONFIG command.
ESMPREF.COMMAND.CONFIG.ACTION	Users who have READ access to this profile have authority for the CONFIG CLEAR ACTION command.
ESMPREF.COMMAND.CONFIG.DATE	Users who have READ access to this profile have authority for the CONFIG CLEAR DATE command.
ESMPREF.COMMAND.CONFIG.EVENT	Users who have READ access to this profile have authority for the CONFIG CLEAR EVENT command.
ESMPREF.COMMAND.CONFIG.IDLE	Users who have READ access to this profile have authority for the CONFIG CLEAR IDLE command.
ESMPREF.COMMAND.CONFIG.GROUP	Users who have READ access to this profile have authority for the CONFIG CLEAR GROUP command.
ESMPREF.COMMAND.CONFIG.MACHINE	Users who have READ access to this profile have authority for the CONFIG CLEAR MACHINE command.
ESMPREF.COMMAND.CONFIG.PAGE	Users who have READ access to this profile have authority for the CONFIG CLEAR PAGE command.
ESMPREF.COMMAND.CONFIG.RULE	Users who have READ access to this profile have authority for the CONFIG CLEAR RULE command.
ESMPREF.COMMAND.CONFIG.SCHEDULE	Users who have READ access to this profile have authority for the CONFIG CLEAR SCHEDULE command.
ESMPREF.COMMAND.CONFIG.SPOOL	Users who have READ access to this profile have authority for the CONFIG CLEAR SPOOL command.
ESMPREF.COMMAND.CP	Users who have READ access to this profile have authority for the CP command.
ESMPREF.COMMAND.DEFACTN	Users who have READ access to this profile have authority for the DEFACTN command.
ESMPREF.COMMAND.DEFDATE	Users who have READ access to this profile have authority for the DEFDATE command.

<i>Table 2. Command-level ESMCLASS profiles (continued)</i>	
<b>Profile</b>	<b>Authority</b>
<i>ESMPREF.COMMAND.DEFDSPS</i>	Users who have READ access to this profile have authority for the DEFDSPS command.
<i>ESMPREF.COMMAND.DEFEMON</i>	Users who have READ access to this profile have authority for the DEFEMON command.
<i>ESMPREF.COMMAND.DEFGROUP</i>	Users who have READ access to this profile have authority for the DEFGROUP command.
<i>ESMPREF.COMMAND.DEFIMON</i>	Users who have READ access to this profile have authority for the DEFIMON command.
<i>ESMPREF.COMMAND.DEFIPCS</i>	USERS who have READ access to this profile have authority for the DEFIPCS command.
<i>ESMPREF.COMMAND.DEFMMON</i>	Users who have READ access to this profile have authority for the DEFMMON command.
<i>ESMPREF.COMMAND.DEFOPTN</i>	Users who have READ access to this profile have authority for the DEFOPTN command.
<i>ESMPREF.COMMAND.DEFPMON</i>	Users who have READ access to this profile have authority for the DEFPMON command.
<i>ESMPREF.COMMAND.DEFRULE</i>	Users who have READ access to this profile have authority for the DEFRULE command.
<i>ESMPREF.COMMAND.DEFSCHD</i>	Users who have READ access to this profile have authority for the DEFSCHD command.
<i>ESMPREF.COMMAND.DEFSMON</i>	Users who have READ access to this profile have authority for the DEFSMON command.
<i>ESMPREF.COMMAND.DEFSRVM</i>	Users who have READ access to this profile have authority for the DEFSRVM command.
<i>ESMPREF.COMMAND.DEFVIEW</i>	Users who have READ access to this profile have authority for the DEFVIEW command.
<i>ESMPREF.COMMAND.DELACTN</i>	Users who have READ access to this profile have authority for the DELACTN command.
<i>ESMPREF.COMMAND.DELACTQ</i>	USERS who have READ access to this profile have authority for the DELACTQ command.
<i>ESMPREF.COMMAND.DELDATE</i>	Users who have READ access to this profile have authority for the DELDATE command.
<i>ESMPREF.COMMAND.DELDSPS</i>	Users who have READ access to this profile have authority for the DELDSPS command.
<i>ESMPREF.COMMAND.DELEMON</i>	Users who have READ access to this profile have authority for the DELEMON command.
<i>ESMPREF.COMMAND.DELGROUP</i>	Users who have READ access to this profile have authority for the DELGROUP command.
<i>ESMPREF.COMMAND.DELIMON</i>	Users who have READ access to this profile have authority for the DELIMON command.
<i>ESMPREF.COMMAND.DELIPCS</i>	USERS who have READ access to this profile have authority for the DELIPCS command.

Table 2. Command-level ESMCLASS profiles (continued)

Profile	Authority
ESMPREF.COMMAND.DELMMON	Users who have READ access to this profile have authority for the DELMMON command.
ESMPREF.COMMAND.DELPMON	Users who have READ access to this profile have authority for the DELPMON command.
ESMPREF.COMMAND.DELRULE	Users who have READ access to this profile have authority for the DELRULE command.
ESMPREF.COMMAND.DELSCHD	Users who have READ access to this profile have authority for the DELSCHD command.
ESMPREF.COMMAND.DELSMON	Users who have READ access to this profile have authority for the DELSMON command.
ESMPREF.COMMAND.DELSRVM	Users who have READ access to this profile have authority for the DELSRVM command.
ESMPREF.COMMAND.DELVIEW	Users who have READ access to this profile have authority for the DELVIEW command.
ESMPREF.COMMAND.LOGTEXT	Users who have READ access to this profile have authority for the LOGTEXT command.
ESMPREF.COMMAND.QUIT	Users who have READ access to this profile have authority for the QUIT command.
ESMPREF.COMMAND.RESUME	Users who have READ access to this profile have authority for the RESUME command.
ESMPREF.COMMAND.REVOKE	Users who have READ access to this profile have authority for the REVOKE command.
ESMPREF.COMMAND.RUN	Users who have READ access to this profile have authority for the RUN command.
ESMPREF.COMMAND.STATUS	Users who have READ access to this profile have authority for the STATUS command.
ESMPREF.COMMAND.STATUS.ACTION	Users who have READ access to this profile have authority for the STATUS DETAIL ACTION command.
ESMPREF.COMMAND.STATUS.ACTIONQ	Users who have READ access to this profile have authority for the STATUS DETAIL ACTIONQ command.
ESMPREF.COMMAND.STATUS.AUTH	Users who have READ access to this profile have authority for the STATUS DETAIL AUTH command.
ESMPREF.COMMAND.STATUS.CONSOLE	Users who have READ access to this profile have authority for the STATUS DETAIL CONSOLE command.
ESMPREF.COMMAND.STATUS.DATASP	Users who have READ access to this profile have authority for the STATUS DETAIL DATASP command.



Table 2. Command-level ESMCLASS profiles (continued)

Profile	Authority
ESMPREF.COMMAND.STATUS.DATE	Users who have READ access to this profile have authority for the STATUS DETAIL DATE command.
ESMPREF.COMMAND.STATUS.EVENT	Users who have READ access to this profile have authority for the STATUS DETAIL EVENT command.
ESMPREF.COMMAND.STATUS.GROUP	Users who have READ access to this profile have authority for the STATUS DETAIL GROUP command.
ESMPREF.COMMAND.STATUS.IDLE	Users who have READ access to this profile have authority for the STATUS DETAIL IDLE command.
ESMPREF.COMMAND.STATUS.IPCS	Users who have READ access to this profile have authority for the STATUS DETAIL IPCS command.
ESMPREF.COMMAND.STATUS.IUCV	Users who have READ access to this profile have authority for the STATUS DETAIL IUCV command.
ESMPREF.COMMAND.STATUS.MACHINE	Users who have READ access to this profile have authority for the STATUS DETAIL MACHINE command.
ESMPREF.COMMAND.STATUS.OPTION	Users who have READ access to this profile have authority for the STATUS DETAIL OPTION command.
ESMPREF.COMMAND.STATUS.PAGE	Users who have READ access to this profile have authority for the STATUS DETAIL PAGE command.
ESMPREF.COMMAND.STATUS.RULE	Users who have READ access to this profile have authority for the STATUS DETAIL RULE command.
ESMPREF.COMMAND.STATUS.SCHEDULE	Users who have READ access to this profile have authority for the STATUS DETAIL SCHEDULE command.
ESMPREF.COMMAND.STATUS.SERVICE	Users who have READ access to this profile have authority for the STATUS DETAIL SERVICE command.
ESMPREF.COMMAND.STATUS.SPOOL	Users who have READ access to this profile have authority for the STATUS DETAIL SPOOL command.
ESMPREF.COMMAND.STATUS.SPOOLUSR	Users who have READ access to this profile have authority for the STATUS DETAIL SPOOLUSR command.
ESMPREF.COMMAND.STATUS.VIEW	Users who have READ access to this profile have authority for the STATUS DETAIL VIEW command.
ESMPREF.COMMAND.STOP	Users who have READ access to this profile have authority for the STOP command.

Table 2. Command-level ESMCLASS profiles (continued)	
Profile	Authority
ESMPREF.COMMAND.SUSPEND	Users who have READ access to this profile have authority for the SUSPEND command.
ESMPREF.COMMAND.VIEWLOG	Users who have READ access to this profile have authority for the VIEWLOG command.

## Command-level profile extensions

If DEFOPTN ESM is set to Y, and AUTHCMD is set to Y, you may also choose to include authorization extensions. Each authorization extension is specified by including the extension number in the DEFOPTN AUTHEXT operand value, such as DEFOPTN AUTHEXT 1 for extension 1.

### Extension 1

Allows the SUSPEND and RESUME commands to be restricted to the resource level. By specifying extension 1 in the AUTHEXT operand, Operations Manager will check the profile below in addition to the profiles from the [Command-level profiles](#).

Table 3. Command-level profile extensions	
Profile	Description
ESMPREF.COMMAND.RESUME.RULE. <i>ruleid</i>	Users who have READ access to this profile have authority to RESUME the RULE <i>ruleid</i> .
ESMPREF.COMMAND.RESUME.SCHEDULE. <i>schdl</i>	Users who have READ access to this profile have authority to RESUME the SCHEDULE <i>schdl</i> .
ESMPREF.COMMAND.RESUME.MACHINE. <i>mmonid</i>	Users who have READ access to this profile have authority to RESUME the MACHINE monitor <i>mmonid</i> .
ESMPREF.COMMAND.RESUME.SPOOL. <i>smonid</i>	Users who have READ access to this profile have authority to RESUME the SPOOL monitor <i>smonid</i> .
ESMPREF.COMMAND.RESUME.EVENT. <i>emonid</i>	Users who have READ access to this profile have authority to RESUME the EVENT monitor <i>emonid</i> .
ESMPREF.COMMAND.RESUME.PAGE. <i>pmonid</i>	Users who have READ access to this profile have authority to RESUME the PAGE monitor <i>pmonid</i> .
ESMPREF.COMMAND.RESUME.IDLE. <i>imonid</i>	Users who have READ access to this profile have authority to RESUME the IDLE monitor <i>imonid</i> .
ESMPREF.COMMAND.SUSPEND.RULE. <i>ruleid</i>	Users who have READ access to this profile have authority to SUSPEND the RULE <i>ruleid</i> .
ESMPREF.COMMAND.SUSPEND.SCHEDULE. <i>schdl</i>	Users who have READ access to this profile have authority to SUSPEND the SCHEDULE <i>schdl</i> .
ESMPREF.COMMAND.SUSPEND.MACHINE. <i>mmonid</i>	Users who have READ access to this profile have authority to SUSPEND the MACHINE monitor <i>mmonid</i> .
ESMPREF.COMMAND.SUSPEND.SPOOL. <i>smonid</i>	Users who have READ access to this profile have authority to SUSPEND the SPOOL monitor <i>smonid</i> .
ESMPREF.COMMAND.SUSPEND.EVENT. <i>emonid</i>	Users who have READ access to this profile have authority to SUSPEND the EVENT monitor <i>emonid</i> .
ESMPREF.COMMAND.SUSPEND.PAGE. <i>pmonid</i>	Users who have READ access to this profile have authority to SUSPEND the PAGE monitor <i>pmonid</i> .

Table 3. Command-level profile extensions (continued)	
Profile	Description
<i>ESMPREF.COMMAND.SUSPEND.IDLE.imonid</i>	Users who have READ access to this profile have authority to SUSPEND the IDLE <i>monitor imonid</i> .

#### Related reference

[Syntax for control commands](#)

[Syntax for Configuration Commands](#)

[DEFOPTN](#)

[DEFVIEW](#)

[VIEWCON](#)



---

## Chapter 3. Creating a startup configuration file

This chapter describes how to create a startup configuration file using the sample file supplied with Operations Manager as a guide.

### Related concepts

[Entering commands](#)

### Related tasks

[Copying the sample configuration file](#)

[Adding comments](#)

[Defining action processing servers](#)

### Related reference

[Using the sample configuration file](#)

[Assigning authority](#)

[Verify the installation and configuration](#)

[Starting Operations Manager and managing the configuration file](#)

After creating the configuration file, you are now ready to start Operations Manager. You can also make changes to the configuration file.

---

## Using the sample configuration file

You can configure Operations Manager by issuing commands directly on the console of the server running GOMMAIN, by issuing commands to the server using the GOMCMD command, or by using a startup configuration file.

Operations Manager provides a sample startup configuration file named OPMGRM1 CONFSAMP. This file includes commands that prepare your Operations Manager environment. You can use the file OPMGRM1 CONFSAMP as a general guide to compose your own startup configuration file. In addition, Operations Manager provides some examples of rules, monitors, and actions in EXAMPLES CONFSAMP. You can use these as a general guide for your own rules, monitors, and actions. For more information about EXAMPLES CONFSAMP, see [Appendix A, “Sample files,” on page 177](#).

After you create your own startup configuration file, specify this file name on the GOMMAIN command in the PROFILE EXEC of OPMGRM1.

The sections that follow describe the possible components of a startup configuration file and shows examples from the startup configuration file OPMGRM1 CONFSAMP.

### Related reference

[Syntax for control commands](#)

[Syntax for Configuration Commands](#)

[GOMMAIN](#)

---

## Copying the sample configuration file

You must copy the sample startup configuration file to the OPMGRM1 198 disk before modifying it. The following steps provide a guideline to copy the sample configuration file:

1. Log on to the VM account your installation will use to modify Operations Manager configuration files. Do *not* use OPMGRM1 or OPMGRS1–4.
2. VMLINK to the minidisk that contains the samples:

```
VMLINK 5697J10F 2C2 < * I RR >
```

3. VMLINK to the minidisk that contains the configuration files:

```
VMLINK OPMGRM1 198 < * E MR >
```

4. Copy the sample file:

```
COPYFILE OPMGRM1 CONFSAMP I filename CONFIG E
```

where *filename* is the name you plan to use for your configuration file.

IBM recommends that you use **OPMGRM1** for the filename. This file can contain one of the following:

- any or all of your configuration information for a single system
- all of the common configuration information for multiple members of a z/VM Single System Image (SSI) cluster.

If you have unique configuration information for each member of an SSI cluster, it can be stored in another configuration file whose filename is the name of the member, and imbedded in this configuration file using the CONFIG statement with the FN &SYSNAME parameter. At Operations Manager startup, &SYSNAME will be replaced with the name of the system.

5. Modify the startup configuration file:

```
XEDIT OPMGRM1 CONFIG E
```

6. Release and detach the minidisk that contains the samples:

```
VMLINK 5697J10F 2C2 < DET >
```

7. Release and detach the minidisk that contains the configuration files:

```
VMLINK OPMGRM1 198 < DET > ( WRITE
```

The following sections describe the possible areas of the configuration file you can modify.

**Related concepts**

[Entering commands](#)

**Related tasks**

[Adding comments](#)

[Defining action processing servers](#)

[Defining system options](#)

**Related reference**

[Assigning authority](#)

[Modifying the startup configuration file](#)

## Adding comments

Operations Manager provides a way to add comments in the startup configuration file. You might do this when describing the purpose of a specific command sequence. To add a comment, place an asterisk (\*) before the comment. For example, if you want to assign authority to a user ID, you might add a comment like this before the AUTH command that assigns the authority:

```
* Define users with control authority to Operations Manager
```

```
AUTH USER user1 ACCESS CONTROL  
AUTH USER user2 ACCESS CONTROL
```

## Assigning authority

Operations Manager provides the AUTH command to assign different types of authority to specific user IDs. These commands are only used if an External Security Manager is *not* in use.

**Related tasks**

[Step 5. \(Optional\) Configure an external security mechanism](#)

## Related reference

[AUTH](#)

## Control authority

To define a specific user ID so that it has authority to use Operations Manager control commands, issue the commands shown below. In this example, `user1` and `user2` are granted control authority.

```
* Define users with control authority to Operations Manager
```

```
AUTH USER MAINT ACCESS CONTROL  
AUTH USER MAINT640 ACCESS CONTROL
```

## Related reference

[Syntax for control commands](#)

## Configuration authority

To give a specific user ID authority to use Operations Manager configuration commands, issue the command shown below. In this example, `user3` is granted configuration authority.

```
* Define users with config authority to Operations Manager.
```

```
AUTH USER MAINT ACCESS CONFIG  
AUTH USER MAINT640 ACCESS CONFIG
```

## Related reference

[Syntax for Configuration Commands](#)

## Console access

To give a specific user ID authority to access all consoles, issue the commands shown below.

**Note:** The commands in these examples give users the ability to view a console and issue commands to that console. For view only authorization, an ESM is required.

```
*Define users who can access all monitored consoles in Operations Manager
```

```
AUTH USER MAINT CONSOLE *  
AUTH USER MAINT640 CONSOLE *
```

To give a specific user ID authority to use the OPERATOR console in Operations Manager, issue the command shown below.

```
* Define users who can access only the OPERATOR console in Operations Manager.
```

```
AUTH USER ADMIN CONSOLE OPERATOR
```

## Spool access

To give a specific user ID authority to access all the spool files on the system using Operations Manager, issue the commands shown below.

**Note:** The commands in these examples give users the ability to view another user's spool files and change the external attributes of those spool files. For view only authorization, an ESM is required.

```
*Define users who can access all spool files on the system using Operations Manager
```

```
AUTH USER MAINT SPOOL *  
AUTH USER MAINT640 SPOOL *
```

To give a specific user ID authority to access the spool files owned by OPERATOR using Operations Manager, issue the command shown below.

```
*Define users who can access only the spool files owned by OPERATOR using
Operations Manager
```

```
AUTH USER ADMIN SPOOL OPERATOR
```

## IP session interaction

To give a specific user ID authority to interact with IP sessions, issue the commands shown below. In this example, user1 is granted authority to interact with the IP sessions named SYS23 and REMOTE1 and user2 is granted authority to interact with all IP sessions.

```
*Define users and the IP sessions they can interact with Operations Manager
```

```
AUTH USER user1 IPCS SYS23
AUTH USER user1 IPCS REMOTE1
AUTH USER user2 IPCS *
```

## Authority assignments with AUTH and external security

If you plan to use an external security mechanism (DEFOPTN ESM Y) for controlling Operations Manager access and authority, IBM recommends assigning authorization with the AUTH command to at least one user. Having a user authorized with AUTH will provide a method to access Operations Manager if the external security mechanism must be disabled temporarily (DEFOPTN ESM N).

```
*Define authority to users for control access
```

```
AUTH USER MAINT ACCESS CONTROL
AUTH USER MAINT640 ACCESS CONTROL
```

**Note:** Refer to [“Step 5. \(Optional\) Configure an external security mechanism”](#) on page 10 for additional information.

## Defining action processing servers

To define service machines to act as action processing servers, issue the commands shown below. In this example, the names of the service machines are OPMGRS1, OPMGRS2, OPMGRS3, and OPMGRS4.

```
* Define service machines to Operations Manager
```

```
DEFSRVM USER OPMGRS1 XAUTOLOG Y
DEFSRVM USER OPMGRS2 XAUTOLOG Y
DEFSRVM USER OPMGRS3 XAUTOLOG Y
DEFSRVM USER OPMGRS4 XAUTOLOG Y
```

### Related reference

[DEFSRVM](#)

## Defining system options

After you authorize users and define action processing servers, you want to define the active options of the Operations Manager system. In the example below, the options instruct Operations Manager to do the following:

- Treat mixed-case input as case sensitive (MIXCASE Y).



- Define the symbols used for VIEWCON, VIEWLOG, VIEWSPL, and CONFIG files. This example sets the forward search command to /, the backward search command to \, the repeat command to =, and the configuration file's continuation symbol as +.

```
*Define options to Operations Manager
```

```
DEFOPTN MIXCASE Y SYMBOLS '/\+=' DSTADJ N
```

You may also define other options that affect various components of Operations Manager. These options include:

- Activate or deactivate ESM.
- Specify attributes to apply to held lines in a user's console.
- Specify how schedules are handled during daylight saving time.
- Specify if command level authorization is used.

### **Related reference**

[DEFOPTN](#)



---

## Chapter 4. Starting Operations Manager and managing the configuration file

After creating the configuration file, you are now ready to start Operations Manager. You can also make changes to the configuration file.

### Related tasks

[Start the Operations Manager service machines](#)

[Creating shortcuts to common commands](#)

IBM provides some sample REXX EXECs as suggested shortcuts to the most common Operations Manager commands. These samples are located on the 5697J10F 2C2 minidisk (or associated SFS directory).

[Managing the configuration file](#)

### Related reference

[Creating a startup configuration file](#)

[Modifying the startup configuration file](#)

[Verify the installation and configuration](#)

---

## Modifying the startup configuration file

After you authorize users, define action processing servers, and define system options, you can include commands in the startup configuration file to create your automated operations environment. The following areas may be defined:

- rules for console message matching
- schedules
- monitors for machines, spool files, and events
- actions to correspond to the above rules, schedules, and monitors
- holidays
- groups

Chapter 8, “Syntax for Configuration Commands,” on page 81 provides a complete list of the configuration commands. Sample commands are also provided in the file EXAMPLES CONFSAMP on the 5697J10F 2C2 minidisk. In addition, the syntax rules for entering commands are described in [Chapter 5, “Entering commands,”](#) on page 31.

---

## Start the Operations Manager service machines

After you modify the startup configuration file, you must modify or verify the GOMMAIN command in the PROFILE EXEC of OPMGRM1 to specify the startup file, and then start the virtual machine. The following steps provide a guideline to modify the GOMMAIN command:

1. Logon to OPMGRM1 or the user ID running GOMMAIN.
2. When the VM READ appears, enter the following so that the PROFILE EXEC is *not* executed:

```
ACC (NOPROF
```

3. Modify the PROFILE EXEC, if needed. For steps to modify the PROFILE EXEC, refer to [“Updates to PROFILE EXEC ”](#) on page 5.
4. Save your changes and exit.
5. Start the virtual machine:

```
#CP IPL CMS
```

6. When VM READ appears, press ENTER.

7. Disconnect from the account running the GOMMAIN command:

```
#CP DISC
```

## Creating shortcuts to common commands

---

IBM provides some sample REXX EXECs as suggested shortcuts to the most common Operations Manager commands. These samples are located on the 5697J10F 2C2 minidisk (or associated SFS directory).

You can copy and modify these samples as appropriate for your installation. Review each of the samples before copying and using it.

Following are the suggested steps for copying these samples to the MAINT 19E disk. If the system is in an SSI cluster, you must copy to the MAINT 19E disk on each member of the cluster.

1. Logon to MAINT $vrm$ , where  $v$  is version,  $r$  is release, and  $m$  is modification, such as MAINT710 for version 7, release 1, modification 0.
2. VMLINK to the minidisk that contains the samples:

```
VMLINK 5697J10F 2C2 < * I RR >
```

3. VMLINK to MAINT 19E minidisk:

```
VMLINK MAINT 19E < 19E F MR >
```

4. Copy the sample files:

```
VMFCOPY VIEW* SAMPEXEC I = EXEC F2 (PRODID 5697J10F%OPMGR OLDDATE REPLACE  
VMFCOPY OMRELOAD SAMPEXEC I = EXEC F2 (PRODID 5697J10F%OPMGR OLDDATE REPLACE  
VMFCOPY VIEW* SAMPPROF I = PROFILE F2 (PRODID 5697J10F%OPMGR OLDDATE REPLACE
```

5. Update the OMRELOAD EXEC to reflect the SSI member names on your system, or remove the member names if the system is not in an SSI cluster.
6. Each of the VIEWxxx PROFILE files will update the default settings for customizable options, such as PF key settings, in the VIEWCON, VIEWSPL, and VIEWLOG commands. Update each of these files as needed for your installation. Refer to the "Usage Notes" for each command for more details.
7. Rebuild the CMS saved system to return the Y-disk (product code or MAINT's 19E disk) to 'shared' status.

```
PUT2PROD SAVECMS
```

8. Log off MAINT $vrm$  to pick up the changes:

```
Logoff
```

## Verify the installation and configuration

---

To test your Operations Manager installation and configuration, follow the instructions in [Appendix G, "Testing Operations Manager installation and configuration,"](#) on page 205.

## Managing the configuration file

---

There are many ways to manage your Operations Manager configuration information once Operations Manager is up and running. IBM suggests the following:

1. From an authorized user, link and access the disk containing the configuration file(s). For example:

```
VMLINK OPMGRM1 198 < * H MR>
```

2. Make and save your changes to the configuration file (OPMGRM1 CONFIG, by default) or any embedded configuration file. Use the CONFIG statement to embed a configuration file within another configuration file.

3. If you have copied the sample OMRELOAD EXEC, issue the following command to have OPMGRM1 re-access the 198 disk and process the changed file:

```
OMRELOAD
```

If you have not copied the sample OMRELOAD EXEC, do the following:

- a. Issue the following command to have OPMGRM1 re-access the disk and see the changed file:

```
GOMCMD OPMGRM1 CMS CMD(ACC 198 E)
```

- b. Reload the configuration file. For example:

```
GOMCMD OPMGRM1 CONFIG FN(OPMGRM1),FT(CONFIG),FM(E),CLEAR ALL
```

4. Release and detach the disk containing the configuration file(s). For example:

```
VMLINK OPMGRM1 198 < DET > ( WRITE
```

### **Related reference**

[CONFIG](#)

[CMS](#)



---

## Chapter 5. Entering commands

Operations Manager requires specific formats when entering commands. Commands consist of a single command and, optionally, one or more operand keywords and their values. Each operand consists of an operand keyword and an operand value.

Below is a basic example of a command input format using parentheses:

```
COMMAND KEYWORD1(operand value),KEYWORD2(operand value),KEYWORD3(operand (value),KEYWORD4('operand (value')
```

Starting with V1R6 of Operations Manager, new syntax rules are applied that eliminate the use of most parentheses. Below is an example of a command input format without the parentheses:

```
COMMAND KEYWORD1 value KEYWORD2 value KEYWORD3 'a value with a space must be limited' KEYWORD4 'value " with a quote'
```

---

### Command syntax rules

**Note:** For compatibility this release of Operations Manager also accepts commands using the syntax of previous releases.

The following rules must be applied when entering commands to properly configure Operations Manager V1.6:

- The COMMAND starts with the first non-blank character.

Below is an example of a command with no operands:

```
STATUS
```

- The minimum operand value length is one character and the maximum operand value length is determined by the command.
- Operand KEYWORD/VALUE pairs may be entered in any order.
- Operand KEYWORD/VALUE pairs may only be entered once.
- If operands are required, one or more spaces must separate the COMMAND from the first operand.
- Spaces are required in between operand keywords.
- One or more blanks separate the operand VALUE and the next operand KEYWORD.
- The operand value follows the operand and may or may not be enclosed in parentheses, as you prefer. If parentheses are used, no spaces are allowed between:
  - the operand and the left parenthesis.
  - the end of the operand value and the right parenthesis.

Below is an example of a command with one operand and parentheses:

```
STATUS DETAIL (RULE)
```

Below is an example of a command with one operand and no parentheses for the operand's value :

```
STATUS DETAIL RULE
```

- If multiple operands are required, you can use spaces to separate the keywords. However, if parentheses are used, a comma must follow the right parenthesis.

Below is an example of a command with multiple operands and using parentheses:

```
DEFRULE NAME (RULE01),MATCH(*HCPMID6001I*),ACTION (MIDNITE)
```

Below is an example of a command with multiple operands using spaces instead of parentheses:

```
DEFRULE NAME RULE01 MATCH *HCPMID6001I* ACTION MIDNITE
```

- Operands may be entered in any order. For example, the following commands are equivalent:

```
DEFRULE NAME(RULE01), MATCH(*HCPMID6001I*), ACTION(MIDNITE)
DEFRULE MATCH(*HCPMID6001I*),NAME(RULE01),ACTION(MIDNITE)
DEFRULE NAME(RULE01), ACTION(MIDNITE), MATCH(*HCPMID6001I*)
DEFRULE NAME RULE01 ACTION MIDNITE MATCH *HCPMID6001I*
```

- If you are using parentheses to delimit the operand value and an operand value contains parentheses, the parentheses must be balanced or single quotes must be used. Below is an example of balanced parentheses in the COMMAND operand:

```
DEFACTN NAME(ACTN01),COMMAND(PRINT A FILE ( NOCC )),ENV (GOM)
```

- If you are using parentheses to delimit the operand value, any operand value may be entered in single quotes (''). Quoting an operand value allows the operand value to bypass the balanced parentheses rule. The rules to quoting are the following:
  - The beginning quote must follow the left parenthesis with no spaces.
  - The operand value begins in the position after the beginning single quote and the operand value ends in the position before the ending quote.
  - The ending quote must be followed by the right parenthesis with no spaces.
  - If a single quote is required in a quoted operand value, use two single quotes to produce a single quote in the operand value.

Below is an example of a quoted operand value (with parentheses) in the COMMAND operand:

```
DEFACTN NAME(ACTN01),COMMAND('PRINT A FILE( NOCC)'),ENV(LVM)
```

- If you are using spaces to delimit the operand keywords and an operand value contains one or more spaces, the operand value must be delimited using single quotes. For example, if the value is only one word, no single quotes are needed:

```
DATA jump
```

However, if the value contains two or more words, single quotes are required:

```
DATA 'jump ship'
```

Below is an example of an operand value (with a space) in the PARM operand:

```
DEFRULE NAME RULE1 MATCH '* HENRY *' ACTION ACTN01 PARM 'Found HENRY'
```

- If you are using spaces to delimit the operand keywords and an operand value is delimited by single quotes, any single quote within the value, such as the apostrophe in "Joe's BBQ", must be entered as two single quotes (' '). The example below shows the MATCH operand with a value containing a space and single quotes.

```
DEFRULE NAME RULEHS MATCH '*Joe's BBQ*' ACTION ACTNHS
```

The next example, "Homer's D'OH Nuts", contains two apostrophes in the text.

```
DEFRULE NAME RULEHS MATCH '*Homer's D'OH Nuts*' ACTION ACTNHS
```

## Where to enter commands

Commands may be entered through one of the following methods:

- *The GOMMAIN command.* The operands of the GOMMAIN command specify a CMS file. This file contains a series of commands which establish the Operations Manager configuration. An Operations Manager



configuration file can be specified and processed by the GOMMAIN command in the PROFILE EXEC of OPMGRM1. This file is commonly called the *startup configuration file*. No authorization checking is performed for commands processed from the startup configuration file. See [“GOMMAIN” on page 35](#) for more information.

- *The GOMCMD command.* The virtual machine where the GOMCMD command is entered must have authorization for the command operand of the GOMCMD command. See [“GOMCMD” on page 36](#) for more information.
- *The VIEWLOG screen.* The input area of the VIEWLOG screen may be used to enter commands. The virtual machine where the VIEWLOG command is entered must have authorization for the command entered in the VIEWLOG screen input area. See [“GOMCMD” on page 36](#) and [“VIEWLOG” on page 159](#) for more information.
- *The SMSG command.* The virtual machine where the SMSG command is entered must have authorization for the command contained in the *msgtext* operand of the SMSG command. See "SMSG" in the *IBM z/VM CP Commands and Utilities Reference* (SC24-6175) for more information.
- *The console of OPMGRM1, or any user ID running the GOMMAIN command.* You must be logged onto the virtual machine running the GOMMAIN command. No authorization checking is performed for commands entered through the console.
- *The CONFIG command.* The FILE, FN, FT, and FM operands of the CONFIG command specify a CMS file. This file contains a series of commands which alter the current Operations Manager configuration. The virtual machine where the CONFIG command is entered must have authorization for each command processed from the file. See [“CONFIG” on page 46](#) for more information.

#### Notes about Operations Manager commands contained in CMS files:

1. The file can have fixed or variable length records with a maximum record length of 255.
2. Any line beginning with an asterisk (\*) is considered a comment and is not parsed by Operations Manager.
3. For improved readability, the commands may be broken into multiple lines joined by a continuation indicator, which is the plus sign (+) by default. The continuation character can be customized with the DEFOPTN command.

Below is an example of a CMS file containing an Operations Manager command broken into multiple lines:

```
*  
*DEFINE A SCHEDULE TO CHECK PRIMARY SPACE ALLOCATION HOURLY  
*  
DEFSCHD NAME SCHD01 +  
  EVERY 01:00 +  
  HOURS 01,02,03,04,05,06,07,18,19,20,21,22,23 +  
  DAYS MTWTF*** +  
  CONFLICT SCHD02,SCHD03 +  
  MAXDEFER 20 +  
  ACTION CHECKSP +  
  PARM PRIMARY
```

#### Related reference

[GOMCMD](#)

[VIEWLOG](#)

[GOMMAIN](#)

[CONFIG](#)



---

## Chapter 6. Syntax for System Management Commands

This chapter describes the Operations Manager system management commands and their syntax.

### Purpose of system management commands

---

Operations Manager provides system management commands that provide control of the Operations Manager environment.

### GOMMAIN

---

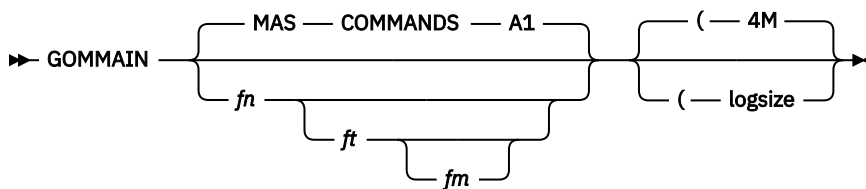


Figure 2. GOMMAIN command syntax

### Authorization

No specific authority is required for this command.

### Purpose

Issue the GOMMAIN command on the main Operations Manager service machine (OPMGRM1) to activate the framework required to run Operations Manager. The GOMMAIN module acts as a central processing service for messages, events, schedules, and monitors. IBM recommends that you issue the GOMMAIN command in the PROFILE EXEC for OPMGRM1. The sample PROFILE EXEC for OPMGRM1 provided with Operations Manager includes this command.

#### Related reference

[Using the sample configuration file](#)

### Operands

#### **fn**

Specifies the one- to eight-character name of the file to be processed. The default is MAS.

#### **ft**

Specifies the one- to eight-character type of the file to be processed. The default is COMMANDS. To specify a file type, the file name operand must also be specified.

#### **fm**

Specifies the one- to two-character mode of the file to be processed. The default is A1. To specify a file mode, the file type operand must also be specified.

#### **logsize**

Specifies the amount of storage used for the log data space storage. The format is *nM* where *n* is between 1 and 2048. The default is 4M.

## Usage notes

1. If no file is specified on the GOMMAIN command and the default file MAS COMMAND A1 does not exist, an error will not occur. Operations Manager runs the command without a file to process.
2. GOMMAIN checks the version, release, and service level of GOMMAIN, GOMSVM, and GOMCMD to ensure compatibility. All of these components must be at the same version, release, and service level to correctly operate.
3. The value &SYSNAME can be used for the *fn* or *ft* operand or both operands. The &SYSNAME value is replaced with the system name.
4. The value &SYSSSIN can be used for the *fn* or *ft* operand or both operands. The &SYSSSIN value is replaced with the name of the SSI cluster.

## Return codes

Table 4 on page 36 lists the return codes for the GOMMAIN command.

Table 4. GOMMAIN return codes	
Return code	Description
0	Finished correctly
-100	Error
-101	An ANCHOR is already active during startup. An attempt was made to run a GOM component in a machine that is already running another GOM component. Re-IPL the virtual machine that received the error. If the problem persists, contact IBM Support.
-102	Unable to establish ANCHOR during startup. Contact IBM Support.
-103	The machine architecture of the virtual machine is not compatible with Operations Manager. GOMMAIN requires virtual machine architecture XC. Use CP SET MACHINE XC to set the required virtual machine architecture. See <i>z/VM CP Commands and Utilities Reference</i> (SC24-6175) for additional information about the SET MACHINE command.
-200	Abend
Other	Return code from a failed z/VM command issued by GOMMAIN. Refer to the error messages for information on the failed command

## GOMCMD

➤ GOMCMD — *userid* — *command* ➤

Figure 3. GOMCMD command syntax

## Authorization

No specific authority is required for this command.

The user issuing GOMCMD must have the authority to issue the specified command. This is accomplished through the Operations Manager AUTH command or through an External Security Manager (ESM). If the user is not authorized to issue the specified command, the user receives an error message. It is suggested that the appropriate users be authorized using an ESM or in the configuration file specified at Operations Manager startup.

### Related tasks

[Step 5. \(Optional\) Configure an external security mechanism](#)

## Related reference

[AUTH](#)

## Purpose

Issue the GOMCMD command to control and monitor Operations Manager. Any valid Operations Manager control or configuration command can be specified.

## Operands

### *userid*

Specifies the user ID of the Operations Manager server that is running GOMMAIN. In a default installation, this is OPMGRM1.

### *command*

Specifies any valid Operations Manager control or configuration command.

## Usage notes

1. GOMCMD checks the version, release, and service level of GOMMAIN, GOMSVM, and GOMCMD to ensure compatibility. All of these components must be at the same version, release, and service level to correctly operate.
2. GOMCMD can be run from an EXEC started by an Operations Manager action.
3. When GOMCMD is used in an EXEC started by an Operations Manager action, you can use &GOMMAIN as the *userid* operand. &GOMMAIN substitutes the VM account running GOMMAIN which started the action. In a default installation, this is OPMGRM1.
4. The return code from GOMCMD is set to the return code of the command that is running.

## Examples

This section contains two examples on how to issue the GOMCMD command. One example shows how to determine the current status of Operations Manager. The second example shows how to determine the current CMSLEVEL.

### Example 1: Status of Operations Manager

If the VM user ID OPMGRM1 is running GOMMAIN, the following command would be issued to determine the current status of Operations Manager:

```
GOMCMD OPMGRM1 STATUS
```

The output from the command is displayed on your screen. An example of the command results is shown in the STATUS [“Examples”](#) on page 77.

### Example 2: Current CMSLEVEL

If the VM user ID OPMGRM2 is running GOMMAIN, the following command would be issued to determine the current CMSLEVEL on OPMGRM2:

```
GOMCMD OPMGRM2 CMS CMD 'QUERY CMSLEVEL'
```

The output from the command is displayed on your screen. An example of the command results is shown below.

## Return codes

[Table 5 on page 38](#) lists the return codes for the GOMCMD command.

Table 5. GOMCMD return codes	
Return code	Description
0	Finished correctly
-100	Error
-101	An ANCHOR is already active during startup. An attempt was made to run a GOM component in a machine that is already running another GOM component. Re-IPL the virtual machine that received the error. If the problem persists, contact IBM Technical Support.
-102	Unable to establish ANCHOR during startup. Contact IBM Technical Support.
-103	The machine architecture of the virtual machine is not compatible with Operations Manager. GOMCMD requires virtual machine architecture ESA or XC. Use CP SET MACHINE ESA or SET MACHINE XC to set the required virtual machine architecture. See <i>z/VM CP Commands and Utilities Reference</i> (SC24-6175) for additional information about the SET MACHINE command.
-104	The GOMCMD component is not compatible with the GOMMAIN or GOMSVM component already active. Ensure the version/service level of GOMCMD is compatible with the version/service level of GOMMAIN and GOMSVM.
-200	Abend
Other	Return code from the command that was passed through GOMCMD.

## GOMSVM

➡ GOMSVM — *userid* ➡

Figure 4. GOMSVM command syntax

### Authorization

No specific authority is required for this command.

### Purpose

Issue the GOMSVM command on the Operations Manager action processing servers to activate the framework required to process actions. The GOMSVM module acts as a service component for GOMMAIN. IBM recommends that you issue the GOMSVM command in the PROFILE EXEC for each action processing server (OPMGRS1 - OPMGRS4 by default.) The sample PROFILE EXEC for OPMGRS<sub>n</sub> provided with Operations Manager includes this command.

### Operands

#### *userid*

Specifies the VM user ID of the Operations Manager server that is running GOMMAIN. In a default installation, this is OPMGRM1.

### Usage notes

GOMSVM checks the version, release, and service level of GOMMAIN, GOMSVM, and GOMCMD to ensure compatibility. All of these components must be at the same version, release, and service level to correctly operate.

# Return codes

Table 6 on page 39 lists the return codes for the GOMSVM command.

Table 6. GOMSVM return codes	
Return code	Description
0	Finished correctly
-100	Error
-101	An ANCHOR is already active during startup. An attempt was made to run a GOM component in a machine that is already running another GOM component. Re-IPL the virtual machine that received the error. If the problem persists, contact IBM Technical Support.
-102	Unable to establish ANCHOR during startup. Contact IBM Technical Support.
-103	The machine architecture of the virtual machine is not compatible with Operations Manager. GOMSVM requires virtual machine architecture ESA or XC. Use CP SET MACHINE ESA or SET MACHINE XC to set the required virtual machine architecture. See <i>z/VM CP Commands and Utilities Reference</i> (SC24-6175) for additional information about the SET MACHINE command.
-200	Abend
Other	Return code from a failed z/VM command issued by GOMSVM. Refer to the error messages for information on the failed command

## GOMTDM

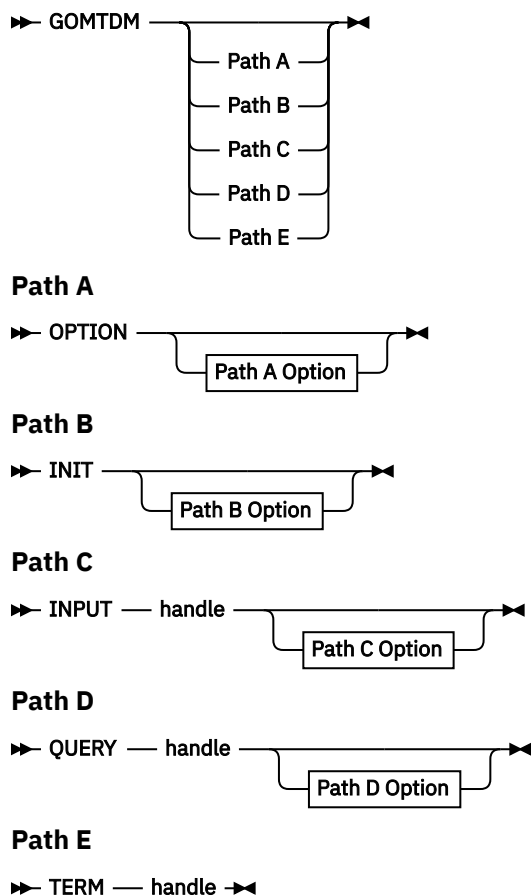


Figure 5. GOMTDM command syntax

## Authorization

No specific authority is required for this command. However, Privilege class G is required for the user ID running GOMTDM. For more information on privilege classes, refer to [Chapter 2, “Configuring Operations Manager,”](#) on page 5.

## Purpose

Issue the GOMTDM command to interact with the terminal device automation infrastructure. The available paths are:

### **GOMTDM OPTION**

Sets options for all sessions. When an option is changed, it effects all sessions. The default options are MSG, TDELAY 0, QDELAY 0, STKFIFO, CCSID 0, NOTRACE, NOWAIT.

### **GOMTDM INIT**

Starts a new session. The handle to the session is returned on the program stack in the order requested.

### **GOMTDM INPUT**

Simulates keyboard input to the session. The handle is the handle returned when the session was started.

### **GOMTDM QUERY**

Allows the program to examine information about the session. The handle is the handle returned when the session was started.

### **GOMTDM TERM**

Terminates a session. The handle is the handle returned when the session was started.

## Operands

### **handle**

Specifies the handle returned when the session was started.

## Path A options

### **MSG|NOMSG**

Controls the error message output. MSG turns on error output to the screen. NOMSG turns off error message output to the screen and suppresses all error messages.

### **TDELAY *n***

Sets the time delay for session output tracing to the screen. After the session input or output data is placed on the screen, the tracing is delayed for *n* seconds.

### **QDELAY *n***

Sets the maximum delay time when querying the session virtual console for a string.

### **STKLIFO|STKFIFO**

Controls the stacking order for values returned on the CMS stack.

### **CCSID *n***

Sets the coded character set identification. This option should be set before the first session is INITed.

### **TRACE|NOTRACE**

Controls the session tracing setting. This option allows all 3270 data streams to be traced to a disk file named LDA TRACE A1.

### **WAIT|NOWAIT**

Controls the behavior of INPUT operations. WAIT causes the processing to delay until the keyboard input inhibited and keyboard locked are inactive. NOWAIT skips the inhibited and locked checks.



## Path B options

### TRACE

Causes all input and output data to be traced to the screen. If you specify this option, your screen size must be defined as a model 4 with 45 lines and 80 columns.

## Path C options

### DATA 'value'

Inputs a data string. The string must be enclosed in single quotes.

### CURSOR *row col*

Simulates moving the cursor to the row and column.

### KEY *keyname*

Simulates a keyboard key. Valid keys are:

- ENTER
- CLEAR
- PA1-PA3
- PF1-PF24
- FT (forward tab)
- BT (backward tab)
- UP (cursor up)
- DOWN (cursor down)
- LEFT (cursor left)
- RIGHT (cursor right)
- NEWLINE
- HOME
- ERASEEOF

## Path D options

### CURSOR

Returns the row and column of the current cursor position on the stack.

### VALUE 'string'

Returns the row and column of the string on the screen on the stack.

- Return code 0 indicates the string was located and the row and column are on the stack.
- Return code 4 indicates the string was *not* found and no value was placed on the stack.

### IMAGEFIL *fn*

Places a copy of the session screen in the CMS file *fn* SCREEN A1.

### VERSION

Returns the build date and time on the stack in the form YYYY/MM/DD hh:mm.

## Usage notes

1. Only one path may be specified on a GOMTDM command.
2. Only one option may be specified on a GOMTDM command.

## Restrictions

1. A maximum of eight (8) simultaneous sessions are allowed from a virtual machine.

## Return codes

Table 7 on page 42 lists the return codes for the GOMTDM command.

Table 7. GOMTDM return codes	
Return code	Description
0	Finished correctly
4	QUERY function did not find the string
8	Syntax or parameter error
12	Abend
16	The machine architecture of the virtual machine is not compatible with Operations Manager. GOMTDM requires virtual machine architecture ESA or XC. Use CP SET MACHINE ESA or SET MACHINE XC to set the required virtual machine architecture. See <i>z/VM CP Commands and Utilities Reference</i> (SC24-6175) for additional information about the SET MACHINE command.
32	Could not set ANCHOR value. Contact IBM Support.
Other	Return code from a failed z/VM command issued by GOMTDM. Refer to the error messages for information on the failed command.

---

# Chapter 7. Syntax for control commands

This chapter describes the Operations Manager control commands and their syntax.

## Purpose of control commands

---

Control commands allow you to control processes relating to Operations Manager. With a control command, you can perform various tasks such as check the status of a system, assign and revoke authorities, and define and delete service machines from Operations Manager's use.

You can type a control command at the console and it runs immediately or you can put control commands into a file so that each time you start Operations Manager, the same checks, authorities, and service machines are in effect and you do not have to reenter them.

Some of these control commands are also valid as configuration commands.

**Related reference**

[Creating a startup configuration file](#)

[Syntax for Configuration Commands](#)

## Authorization for control commands

---

You must use an ESM or the AUTH command to give a specific user the authority to issue control commands. To give users access to specific control commands, an ESM is required.

**Related tasks**

[Step 5. \(Optional\) Configure an external security mechanism](#)

## AUTH

---

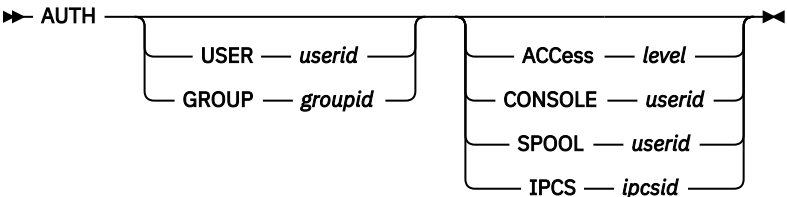


Figure 6. AUTH command syntax

## Authorization

The user issuing this command must be authorized for control commands.

## Purpose

Use the AUTH command to give authorization to a user or group to issue Operations Manager commands, view consoles, view spooled output, and interact with IP sessions.

## Operands

**USER** *userid*

Identifies the user being given authorization.

**GROUP** *groupid*

Identifies the group being given authorization.

**ACCEss level**

Specifies the level of Operations Manager commands being authorized for the user or group. The *level* may be CONTROL or CONFIG.

**CONSOLE userid**

Specifies that the user or group is authorized to view and interact with the console belonging to *userid* or a combined view named *userid*. *userid* may be a wildcard pattern.

**SPOOL userid**

Specifies that the user or group is authorized to view spooled output belonging to *userid*. *userid* may be a wildcard pattern.

**IPCS ipcsid**

Specifies that the user or group is authorized to interact with the IP session defined as *ipcsid*. The *ipcsid* variable may be a wildcard pattern.

**Related concepts**

[Wildcard pattern matching](#)

Wildcard pattern matching is used by several of the commands. Wildcards allow you to search for and process characters that are similar.

**Related tasks**

[Step 5. \(Optional\) Configure an external security mechanism](#)

**Related reference**

[DEFGROUP](#)

[DEFOPTN](#)

## Operand value lengths

Table 8 on page 44 lists the operand value lengths for the AUTH command.

Table 8. AUTH operand value lengths	
Operand	Length
ACCESS	6-7
CMDIPCS	1-8
CONSOLE	1-8
GROUP	1-8
USER	1-8
SPOOL	1-8

## Examples

A couple of examples are provided to help you see how to use the AUTH command and the typical results that the AUTH command generates.

**Example 1: Assign group SYSADM authority for all Operations Manager functions.**

```
OPMGRM1 CONFIG E2 V 80 Trunc=80 Size=71 Line=18 Col=1 Alt=0
*
* Authorize the SYSADM group for all Operations Manager functions
*
AUTH GROUP SYSADM ACCESS CONTROL
AUTH GROUP SYSADM CONSOLE *
AUTH GROUP SYSADM SPOOL *
AUTH GROUP SYSADM IPCS *
*
```

Figure 7. AUTH command with GROUP SYSADM operands

## Example 2: Assign specific authority to individual users (MAINT640 and TS3163).

```
OPMGRM1 CONFIG E2 V 80 Trunc=80 Size=71 Line=18 Col=1 Alt=0
*
* Authorize the SYSADM group for all Operations Manager functions
*
AUTH USER MAINT640 ACCESS CONTROL
AUTH USER TS3163 ACCESS CONTROL
AUTH USER TS3163A ACCESS CONTROL
AUTH USER MAINT640 ACCESS CONFIG
AUTH USER TS3163 ACCESS CONFIG
AUTH USER TS3163A ACCESS CONFIG
AUTH USER MAINT640 CONSOLE(*)
AUTH USER TS3163 CONSOLE(*)
AUTH USER TS3163A
CONSOLE(*)

*
```

Figure 8. AUTH command with USER operand

## Usage notes

1. Either USER or GROUP must be specified, but not both. USER and GROUP are mutually exclusive.
2. One of the following operands must be specified: ACCESS, CONSOLE, SPOOL, or IPCS.
3. For an explanation of wildcard pattern matching, see [Appendix B, “Wildcard pattern matching,” on page 181](#).
4. When a user or group is authorized for access level CONTROL, they are also authorized for access level CONFIG when one of the following is also true:
  - a. ESM is set to N on the DEFOPTN command, or
  - b. ESM is set to Y and AUTHCMD is set to N on the DEFOPTN command.
5. A user authorized for CONTROL is *not* authorized for CONFIG if ESM is set to Y and AUTHCMD is set to Y on the DEFOPTN command.

## Return codes

Table 9 on page 45 lists the return codes for the AUTH command.

Table 9. AUTH return codes	
Return code	Description
0	Finished correctly
-100	Error

## CMS

➡ CMS — CMD — *command* ➡

Figure 9. CMS command syntax

## Authorization

The user issuing this command must be authorized for control commands.

## Purpose

Use the CMS command to present a command to CMS for processing on the main Operations Manager server (OPMGRM1) and display the results.

# Operands

## CMD command

Specifies the CMS command to process. The value may be entered as command.

# Operand value lengths

Table 10 on page 46 lists the operand value lengths for the CMS command.

Table 10. CMS operand value lengths	
Operand	Length
CMD	1-120

# Examples

Issue the CMS command to determine the current CMSLEVEL on OPMGRM1.

## Example 1: Issue the CMS command to see the current CMSLEVEL.

```
GOMCMD OPMGRM1 CMS CMD 'QUERY CMSLEVEL'
08/29/19 06:12:48 GOMCMD0210I  CMS Level 28, Service Level 1901
Ready; T=0.01/0.01 06:12:48
```

Figure 10. Example of CMS command

# Usage notes

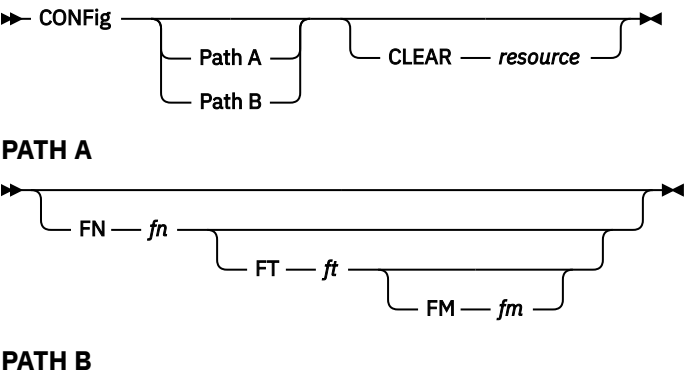
None

# Return codes

Table 11 on page 46 lists the return codes for the CMS command.

Table 11. CMS return codes	
Return code	Description
0	Finished correctly
-100	Error
Other	Error code from a called interface, command, or service

# CONFIG



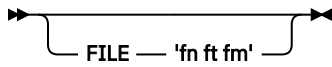


Figure 11. CONFIG command syntax

## Authorization

The user issuing this command must be authorized for configuration or control commands. If CONFIG is entered via the GOMCMD interface, SMSG interface, or VIEWLOG command line, each command in the specified "config" file is checked for authorization.

## Purpose

Use the CONFIG command to process the contents of a file containing Operations Manager commands.

## Operands

### FN *fn*

Specifies the file name of the file to be processed. The default is MAS.

### FT *ft*

Specifies the file type of the file to be processed. The default is COMMANDS.

### FM *fm*

Specifies the file mode of the file to be processed. The default is A1. This is the file mode on OPMGRM1.

### FILE '*fn ft fm*'

Specifies the file name, file type, and file mode to be processed. The default is MAS COMMANDS A1.

### CLEAR *resource*

Indicates that all existing instances for the specified resource type are removed before processing the specified file. The default is to not remove any resource. Multiple resources may be specified in a comma-separated list.

The valid resources are:

- ALL (clears all of the following resources)
- RULE
- MACHINE
- SCHEDULE
- ACTION
- GROUP
- DATE
- SPOOL
- EVENT
- PAGE
- IDLE

## Operand value lengths

Table 12 on page 47 lists the operand value lengths for the CONFIG command.

Table 12. CONFIG operand value lengths	
Operand	Length
CLEAR	1-120

Table 12. CONFIG operand value lengths (continued)	
Operand	Length
FILE	1-20
FM	1-2
FN	1-8
FT	1-8

## Example

An example is provided to help you understand how to use the CONFIG command to process the contents of a file containing Operations Manager commands, and to see the typical results of this command.

### Example 1: Use the CONIG command to clear the contents of the current Operations Manager configuration before reloading any new entries from the specified configuration file.

An example of the CONFIG command is shown below:

```
GOMCMD OPMGRM1 CONFIG FILE 'OPMGRM1 CONFIG E' CLEAR ALL
```

This command clears all available resources that are currently loaded, and reloads the main configuration file.

## Usage notes

1. The CONFIG command can be used within a configuration file containing Operations Manager commands, allowing you to embed other configuration files. You can nest the CONFIG command within included files. There is no restriction on the depth of nested files. You can use this feature in the startup configuration file for the GOMMAIN command to organize your configuration data into multiple files.
2. If the FT operand is entered, the FN operand must be entered.
3. If the FM operand is entered, the FT operand must be entered.
4. The operands on one path are mutually exclusive with operands on other paths.
5. Use &SYSNAME to substitute the current system name for either the *fn* or *ft* value, or both values, in the FN, FT, and FILE operands. The &SYSNAME variable is replaced with the system name.
6. Use &SYSSIN to substitute the current SSI cluster for either the *fn* or *ft* value, or both values, in the FN, FT, and FILE operands. The &SYSSIN variable is replaced with the system SSI cluster name.

## Return codes

Table 13 on page 48 lists the return codes for the CONFIG command.

Table 13. CONFIG return codes	
Return code	Description
0	Finished correctly
-100	Error

## CP

➡ CP — CMD — *command* ➡

Figure 12. CP command syntax



# Authorization

The user issuing this command must be authorized for control commands.

# Purpose

Use the CP command to present a command to the Control Program (CP) for processing on OPMGRM1, and display the results.

# Operands

**CMD command**  
Specifies the CP command to process.

# Operand value lengths

Table 14 on page 49 lists the operand value lengths for the CP command.

Table 14. CP operand value lengths	
Operand	Length
CMD	1-120

# Examples

Issue the CP command to query details of Operations Manager.

## Example 1: Use the CP command to display (QUERY) the USERID details.

```
GOMCMD OPMGRM1 CP CMD 'QUERY USERID'
08/30/19 06:03:26 GOMCMD0206I OPMGRM1 AT ZVM6KT02
Ready; T=0.01/0.01 06:03:26
```

Figure 13. Example of CP command with USERID operand

## Example 2: Use the CP command to display (QUERY) the CPLEVEL details, such as version number and service level, currently being used.

```
GOMCMD OPMGRM1 CP CMD 'QUERY CPLEVEL'
08/30/19 02:07:03 GOMCMD0206I z/VM Version 6 Release 4.0, service level 1901 (64-
bit)
08/30/19 02:07:03 GOMCMD0206I Generated at 07/30/19 14:41:27 CDT
08/30/19 02:07:03 GOMCMD0206I IPL at 07/30/19 16:13:18 CDT
Ready; T=0.01/0.01 06:04:31
```

Figure 14. Example of CP command with CPLEVEL operand

## Example 3: Use the CP command to display (QUERY) the USERS details, such as number of users.

```
GOMCMD OPMGRM1 CP CMD 'QUERY USERS'
08/30/19 06:06:01 GOMCMD0206I 23 USERS, 0 DIALED, 0 NET
Ready; T=0.01/0.01 06:06:01
```

Figure 15. Example of CP command with USERS operand

# Usage notes

None

# Return codes

Table 15 on page 50 lists the return codes for the CP command.

Table 15. CP return codes	
Return code	Description
0	Finished correctly
-100	Error
Other	Error code from a called interface, command, or service

# DEFDSPS

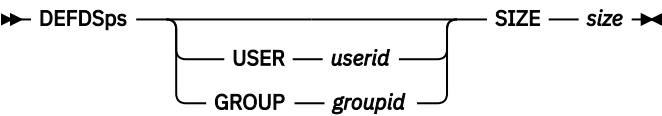


Figure 16. DEFDSPS command syntax

# Authorization

The user issuing this command must be authorized for control commands.

# Purpose

Use the DEFDSPS command to define the amount of virtual storage or memory used to capture the data for a monitored user's console or combined view. When the virtual storage for the user's data becomes full, the oldest data will be removed to make room for the most recent data. In addition, all data for a monitored user's console is written to the Operations Manager log file on disk.

# Operands

## USER *userid*

Specifies a user name, a combined view, or a wildcard pattern. Output collected for the user, the combined view, or any user name or combined view matching the wildcard pattern is allocated the amount of storage defined by the SIZE operand.

## GROUP *groupid*

Specifies a group name. Output collected for any user or combined view that is a member of the group is allocated the amount of storage defined by the SIZE operand.

## SIZE *size*

Specifies the number of bytes or megabytes needed to create the data space that is used to contain the output of monitored users. The valid range is 1-1073741824. The input is rounded up to the nearest 1048576.

If megabytes (MB) are used, the size must be suffixed by M. The startup default is 1048576 (1M) and is for all users.

## Related concepts

[Wildcard pattern matching](#)

Wildcard pattern matching is used by several of the commands. Wildcards allow you to search for and process characters that are similar.

## Related reference

[DEFGROUP](#)

[DEFVIEW](#)

## Operand value lengths

Table 16 on page 51 lists the operand value lengths for the DEFDSPS command.

Table 16. DEFDSPS operand value lengths	
Operand	Length
SIZE	1-10
USER	1-8
GROUP	1-8

## Examples

A few examples are provided to help you understand how to use the DEFDSPS command and to show the typical results of this command.

### Example 1: Define 5M of virtual storage or memory to capture the console data for the monitored user MAINT640.

```
GOMCMD OPMGRM1 DEFDSPS USER MAINT640 SIZE 5M
Ready; T=0.01/0.01 04:26:45
```

Figure 17. DEFDSPS command issued for user

### Example 2: Define 10M of virtual storage to capture the console data for the monitored group ROCKET1.

```
GOMCMD OPMGRM1 DEFDSPS GROUP ROCKET1 SIZE 10M
Ready; T=0.01/0.01 02:22:45
```

Figure 18. DEFDSPS command issued for group

## Usage notes

1. Either USER or GROUP must be specified.
2. USER and GROUP are mutually exclusive.
3. For an explanation of wildcard pattern matching, see [Appendix B, “Wildcard pattern matching,” on page 181](#).
4. If you have multiple DEFDSPS commands, the order of the commands is important; each occurrence does not override the previous command. In the following example, the second command does not override the first command:

```
DEFDSPS USER USER1 SIZE 2M
DEFDSPS USER USER* SIZE 3M
DEFDSPS USER * SIZE 1M
```

In this example, USER1's console has a 2M data space, the consoles of all other user IDs starting with USER have a 3M data space, and all other consoles have a 1M data space.

5. The processing of a DEFDSPS command does not allocate a data space. A data space is allocated when the initial input from a monitored user's console is received by Operations Manager.
6. The maximum number of data spaces and the maximum amount of storage for all data spaces allocated by Operations Manager is limited by the CP directory entry for the user ID running GOMMAIN, which is OPMGRM1 by default. The CP directory's XCONFIG ADDRSPACE statement controls the maximum number of data spaces and the maximum amount of storage for all data spaces. If a large number of data spaces are expected to exceed 16MB in size, examine the CP directory XCONFIG ADDRSPACE statement and adjust the TOTSIZE parameter as needed.

## Return codes

Table 17 on page 52 lists the return codes for the DEFDSPS command.

Table 17. DEFDSPS return codes	
Return code	Description
0	Finished correctly
-100	Error

## DEFIPCS

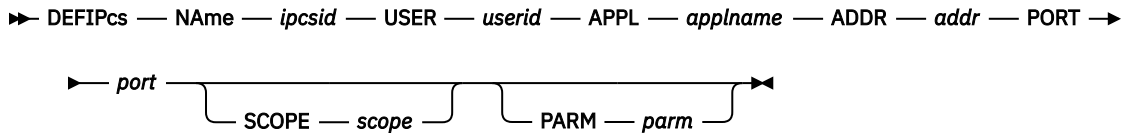


Figure 19. DEFIPCS command syntax

## Authority

The user issuing this command must be authorized for control commands.

## Purpose

Use the DEFIPCS command to define an IP communication session. IP communication sessions in Operations Manager can be used to:

- Send messages or data to another system via an IP connection.
- Receive messages from another system via an IP connection, and process those messages through Operations Manager rules.
- Receive syslog messages from another system and process those messages through Operations Manager rules.
- Run actions (triggered locally) on other LPARs running Operations Manager.
- Provide name services lookup.

## Operands

### **Name** *ipcsid*

Specifies the name of the IP session.

### **USER** *userid*

Specifies the name of the VM *userid* providing the IP communications support. On most z/VM systems, this user ID is TCPIP.

### **APPL** *applname*

Specifies the name of the Operations Manager processing program. See the “Usage notes” on page 53 for available processing applications.

### **ADDR** *varname*

Specifies the address or hostname for the connection. The address format is one of the following:

- The format for IPV4 addresses is a dotted decimal notation, *n . n . n . n*.
- The format for IPV6 addresses is a hex notation, *hhhh : hhhh : . . . : hhhh*
- If the operand value does not contain an IPV4 or IPV6 address, the value of *varname* is passed to name services for IP address lookup. The host name must be a fully qualified name. A name

services IP connection must be defined using DEFIPCS with an APPL value of GOMRSNT. It is recommended that DEFIPCS APPL GOMRSNT be defined first to decrease DNS lookup time.

For APPL values of GOMRSYL, GOMRSYM, GOMRSIF, and GOMRSCL, the value of *varname* in ADDR *varname* refers to the local system. This is often specified as a null address (0.0.0.0 for IPV4 or :: for IPV6) to indicate Operations Manager will accept inbound messages from any IP interface defined for the local system. If the address or host name of an IP interface is entered, only messages from that particular IP interface are accepted.

For APPL values of GOMRSGW, GOMRSNT and GOMRSCC, the value of *varname* in ADDR *varname* refers to a remote system. For APPL GOMRSNT, which is the name lookup support, the ADDR must be an IPV4 or IPV6 value. A hostname is not allowed.

**PORT *port***

Specifies the port for the connection.

- For APPL values of GOMRSYL, GOMRSYM, GOMRSIF, and GOMRSCL, the value of *port* in PORT *port* refers to the local system.
- For APPL values of GOMRSGW, GOMRSNT, and GOMRSCC, the value of *port* in PORT *port* refers to a remote system.

**SCOPE *scope***

Specifies the address scope for IPV6 networks requiring a scope value.

**PARM *parm***

Specifies the parameter passed to the processing program. The value may be entered as '*parm*'.

## Operand value lengths

The table below lists the operand value lengths for the DEFIPCS command.

Table 18. DEFIPCS operand value lengths	
Operand	Length
ADDR	2-253
APPL	8-8
NAME	1-8
PARM	1-120
PORT	1-5
SCOPE	1-5
USER	1-8

## Examples

To help you understand how to use this command, several examples are provided in Appendix F, "Using internet protocol and networked components," on page 197. You can also refer to the EXAMPLES CONFSAMP file on the 5697J10F 2C2 (samples) disk for more examples.

## Usage notes

1. The SCOPE operand is only used with IPV6 addresses.
2. The APPL operand can specify any of the following processing applications:

**GOMRSIF**

Provides support to process CMS file input from a local or remote z/VM system, adds the file content to a console view and the Operations Manager log, and filters against rules. No PARM is used.

## GOMRSYL

Provides support to process input from a remote *syslog* transmitter using UDP protocol. Messages are added to a console view, the Operations Manager log, and filtered against rules. The PARM operand describes the following fields:

Table 19. PARM Operand fields	
Position	Description
01 - 08	Identifies the USER that is to be passed for rule processing. If you specify blanks, the internal program name is used.
09 - 16	Specifies the filename of the ASCII to EBCDIC translation table. The filetype is TABLE. This file may be located on any accessed disk. If you specify blanks, the internal ASCII to EBCDIC table is used.
17 - 20	Identifies which character encoding scheme to use: ISO8 or UTF8. If you specify blanks, ISO8 is used.
21	<p>Indicates whether to display the sender's address (Y, S, or N).</p> <ul style="list-style-type: none"><li>• A value of "Y" displays the address in a fixed, full format.</li><li>• A value of "S" displays the address in a shorthand format.</li><li>• A value of "N" does not display the address. If you specify a blank, N is used.</li></ul> <p>For example, a shorthand format of an IPV4 address is:</p> <pre>10.2.132.3</pre> <p>An example of a full address for IPV4 is:</p> <pre>010.002.132.003</pre> <p>For IPV6, an example of a shorthand format is:</p> <pre>2001:db8::2:1</pre> <p>An example of a full address for IPV6 is:</p> <pre>2001:0db8:0000:0000:0000:0000:0002:0001</pre>

## GOMRSYM

Provides support to process input from a remote *syslog* transmitter using TCP protocol. Messages are added to a console view, the Operations Manager log, and filtered against rules. The data is expected to follow the octet counting frame format of RFC6587. The PTF for APAR PI74064 is required to use this function.

The PARM operand describes the following fields:

Table 20. PARM Operand fields	
Position	Description
01 - 08	Identifies the USER that is to be passed for rule processing. If you specify blanks, the internal program name is used.
09 - 16	Specifies the filename of the ASCII to EBCDIC translation table. The filetype is TABLE. This file may be located on any accessed disk. If you specify blanks, the internal ASCII to EBCDIC table is used.
17 - 20	Identifies which character encoding scheme to use: ISO8 or UTF8. If you specify blanks, ISO8 is used.

Table 20. PARM Operand fields (continued)	
Position	Description
21	<p>Indicates whether to display the sender's address (Y, S, or N).</p> <ul style="list-style-type: none"> <li>• A value of "Y" displays the address in a fixed, full format.</li> <li>• A value of "S" displays the address in a shorthand format.</li> <li>• A value of "N" does not display the address. If you specify a blank, N is used.</li> </ul> <p>For example, a shorthand format of an IPV4 address is:</p> <pre>10.2.132.3</pre> <p>An example of a full address for IPV4 is:</p> <pre>010.002.132.003</pre> <p>For IPV6, an example of a shorthand format is:</p> <pre>2001:db8::2:1</pre> <p>An example of a full address for IPV6 is:</p> <pre>2001:0db8:0000:0000:0000:0000:0002:0001</pre>

#### GOMRSGW

Provides support for a general writer. The PARM operand describes the protocol to be used for the connection.

To use this connection to send data to another system, refer to [“CMDIPCS” on page 150](#).

Table 21. PARM Operand fields	
Position	Description
01 - 03	Identifies the protocol to be used for the connection: TCP or UDP. If you specify blanks, TCP is used.
04	Indicates whether a secure connection is required (Y or N). If this field is blank, N is used. To require a secure connection, the protocol must be TCP, and z/VM must include support for dynamic TLS APIs.
05-12	<p>Specifies additional information if a secure connection is used:</p> <ul style="list-style-type: none"> <li>• If a secure connection is being used, this identifies the label of the certificate for establishing the secure connection.</li> <li>• If a secure connection is not being used, this field is ignored.</li> </ul>

#### GOMRSNT

Provides support for name services lookup. The PARM operand describes the protocol to be used for the connection. Valid values are TCP or UDP (in position 01 - 03). If you specify blanks or omit the PARM operand, TCP will be used.

#### GOMRSCC

Defines a connection to another z/VM system, in a remote execution environment, running Operations Manager, to which you want to send actions for execution. PARM describes the following fields.

<i>Table 22. PARM Operand fields</i>	
<b>Position</b>	<b>Description</b>
01	Indicates whether a secure connection is required (Y or N). If this field is blank, N is used. To require a secure connection, the protocol must be TCP, and z/VM must include support for dynamic TLS APIs.
02-09	Specifies additional information if a secure connection is used: <ul style="list-style-type: none"> <li>• If a secure connection is being used, this identifies the label of the certificate for establishing the secure connection.</li> <li>• If a secure connection is not being used, this field is ignored.</li> </ul>

## **GOMRSC**

Defines a connection to listen for requests from other z/VM systems, in a remote execution environment, running Operations Manager. These requests are to run an action on the local z/VM system.

<i>Table 23. PARM Operand fields</i>	
<b>Position</b>	<b>Description</b>
01	Indicates whether a secure connection is required (Y or N). If this field is blank, N is used. To require a secure connection, the protocol must be TCP, and z/VM must include support for dynamic TLS APIs.
02-09	Specifies additional information if a secure connection is used: <ul style="list-style-type: none"> <li>• If a secure connection is being used, this identifies the label of the certificate for establishing the secure connection.</li> <li>• If a secure connection is not being used, this field is ignored.</li> </ul>

## **Return codes**

The table below lists the return codes for the DEFIPCS command.

<i>Table 24. DEFIPCS return codes</i>	
<b>Return code</b>	<b>Description</b>
0	Finished correctly
-100	Error



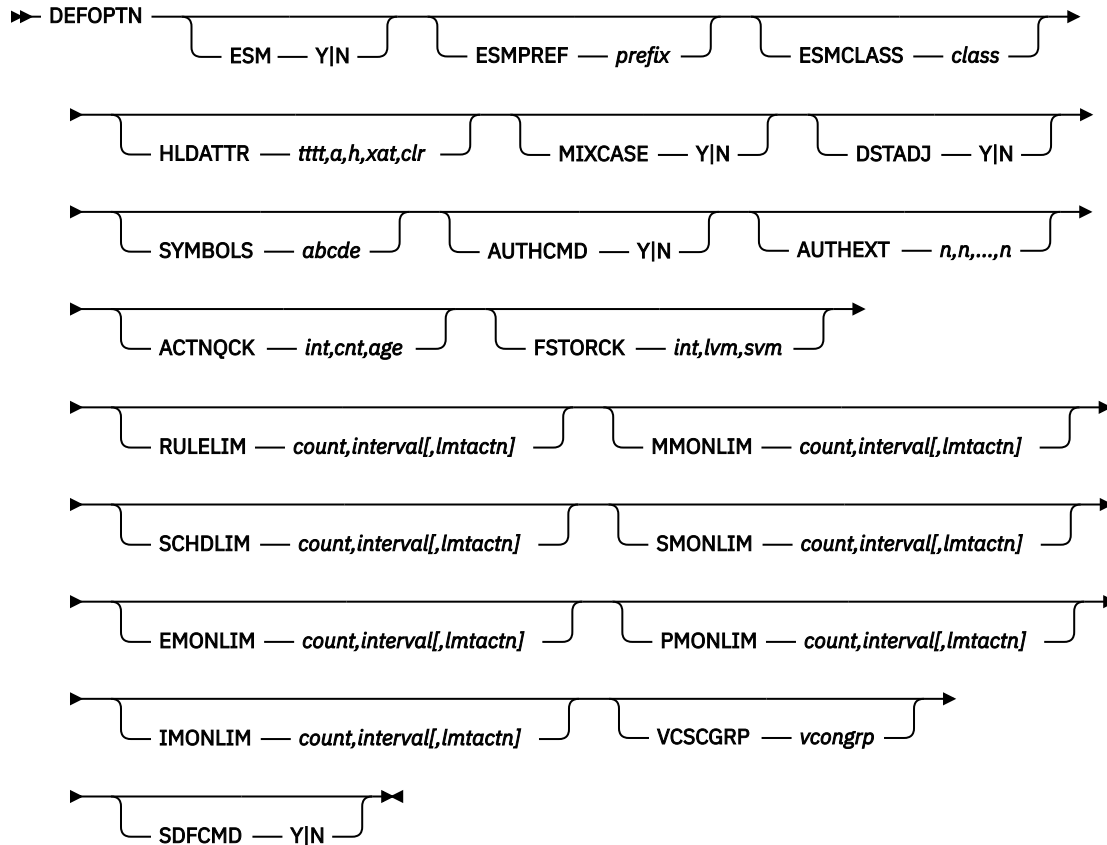


Figure 20. DEFOPTN command syntax

## Authorization

The user issuing this command must be authorized for control commands.

## Purpose

Use the DEFOPTN command to change the active options.

## Operands

### ESM Y|N

Activates or deactivates the use of an external security manager to check for authorization. The startup default is N.

### ESMCLASS class

Specifies the class which contains the external security manager profiles for this instance of Operations Manager. The startup default is FACILITY.

### ESMPREF prefix

Specifies the prefix for the external security manager profiles for this instance of Operations Manager.

- (Default) If no value is specified, Operations Manager looks for profiles with a prefix of RKT . OPM without ' GOM . ' preceding the prefix.
- If you specify a value for ESMPREF, Operations Manager looks for profiles with a prefix of GOM . *esmprefix*. For example, if you specify ESMPREF OPSMGR in the Operations Manager configuration file, Operations Manager will look for profiles with the prefix GOM . OPSMGR.

**HLDATTR *tttt,a,h,xat,clr***

Specifies a time and a set of attributes to apply to "held" lines in a user's console. The startup default is 0060,N,Y,AAI,CAI.

- *tttt* — Specifies a number of seconds. Must be a four-digit number. The default is 60 seconds, which is entered as 0060.

If a "held" line in a user's console has not been responded to within the number of seconds specified, the next four attributes (*a, h,, xat, clr*) are applied to the line.

- *a* — Specifies if the audible alarm is sounded. Values are Y or N. The default is N.
- *h* — Specifies if the line is highlighted. Values are Y or N. The default is Y.
- *xat* — Specifies the extended attributes applied to the line. The default is AAI. Valid values are:
  - AAI – Leave as-is.
  - ABL – Blinking
  - ARV – Reverse video
  - AUL – Underline
- *clr* — Specifies the color applied to the line. The default is CAI. Valid values are:
  - CAI – Leave as-is.
  - CBL – Blue
  - CRE – Red
  - CPI – Pink
  - CGR – Green
  - CCY – Cyan
  - CYE – Yellow
  - CWH – White

**MIXCASE *Y/N***

Specifies if mixed-case data is treated as case-sensitive data. This option affects command input and user console input handling. The default is Y. This operand is used when:

- Text matching is specified for rules.
- Input commands are entered from the OPMGRM1 console or when using the GOMCMD command interface.
- Input commands are listed in a file, such as the startup configuration file, or specified on the CONFIG command.
- Text search operations are executed while viewing the results from the VIEWLOG, VIEWCON, or VIEWSPPL commands.

To use case-sensitive data, the value for MIXCASE must be set to 'Y'. For example, when DEFACTN uses a COMMAND operand that is processed through the CP diagnose x'0008' interface, case-sensitive data may be sent to z/VM guests, such as Linux®. However, this data will only be sent as mixed case if the DEFOPTN value for MIXCASE is set to 'Y'. For more information on DEFACTN and the COMMAND operand, refer to [“Usage notes” on page 88](#).

**DSTADJ *Y/N***

Specifies if the schedule processing time is adjusted when a change in daylight savings time is detected. The startup default is Y.

- DSTADJ Y causes Operations Manager to trigger schedules (defined with DEFSCHD) strictly by wall clock time.

When a time zone change causes the local time to move forward, schedules, which would have occurred within the period of time represented by the time zone change, are not executed during the time change. They are executed at their next scheduled execution time.

When a time zone change causes the local time to move backward, schedules, which should occur within the period of time represented by the time zone change, are executed twice since the specified time occurs twice.

- **DSTADJ N** causes Operations Manager to trigger schedules (defined with DEFSCHD) based on the time zone change.

When a time zone change causes the local time to move forward, schedules, which would have been eligible for execution within the period of time represented by the time zone change, are immediately scheduled for execution at the start of the time change.

When a time zone change causes the local time to move backward, schedules, which are eligible for execution within the period of time represented by the time zone change, are executed only once. They are not repeated.

### **SYMBOLS *abcde***

Specifies the symbols to use in various components. Each symbol is one character and is positional in the list. If you do not wish to change a symbol, you must input a space. The value may be entered **abcde**.

The symbols in the list and the defaults are:

- *a* — Forward search command character in VIEWCON, VIEWLOG, and VIEWSPL. The default is a forward slash (/).
- *b* — Backward search command character in VIEWCON, VIEWLOG, and VIEWSPL. The default is a backward slash (\).
- *c* — Repeat command character. The default is the equal sign (=).
- *d* — Line continuation character. The default is a plus sign (+).
- *e* — Masking character for messages in the Operations Manager log. The default is an asterisk (\*).

### **AUTHCMD *Y/N***

Specifies that command-level authorization should be used instead of the CONTROL and CONFIG level authorization. If command-level authorization is used, ESM authorization checking must be active.

Specifying AUTHCMD *Y* activates use of the standard command-level profiles. See the [“ESM class and Facility class profiles”](#) on page 11 for the list of standard command level profiles.

If AUTHCMD *Y* is specified without also specifying ESM *Y*, it has no effect.

The default is *N*.

### **AUTHEXT *n,n,...,n***

Specifies which of the *extended* command level profile sets (1 - 32) should be used in addition to the standard command level profiles. The default is 0, meaning no authorization extensions are used, only the standard profiles.

Only the profile sets that are listed in the AUTHEXT command are activated. Use AUTHEXT 0 to deactivate all extended command level sets.

Normally when you are using the DEFOPTN command with an ESM and ESM command authorization, it would be a command like this:

```
DEFOPTN ESM Y AUTHCMD Y
```

If you wish to include the additional functionality provided with extended command level sets, you must add the AUTHEXT command followed by the number of the extended command-level set(s) you want to use. You may choose one or more of the following:

Table 25. Extended command levels	
Extended command level set	Additional Function
1	Authorize SUSPEND/RESUME at the resource level. See <a href="#">Table 3 on page 18</a> for a list of profiles.

Table 25. Extended command levels (continued)	
Extended command level set	Additional Function
2 - 32	To be determined

Each extension profile set corresponds to a number from 1 - 32. Only the extension profile sets specified in the AUTHEXT command are activated.

You can choose which extension(s) you want to use in your configuration. For example, to use authorization extensions 2 and 4, but not 1 or 3, you would specify this command:

```
DEFOPTN ESM Y AUTHCMD Y AUTHEXT 2,4
```

If AUTHEXT is issued without also including ESM Y and AUTHCMD Y, it has no effect. If AUTHEXT specifies an extended command level profile set that is not supported, it has no effect.

The startup default is AUTHEXT 0.

#### **ACTNQCK *int,cnt,age***

Specifies how the action processing queue should be monitored. The queue is checked every *int* minutes.

A message to the system operator is issued if the queue contains more than *cnt* elements or contains an element over *age* minutes old. Each value for *int*, *cnt*, and *age* must be a four-digit number. If *int*, *cnt*, or *age* are zero, the corresponding checks are not performed.

The startup default is 0015,0100,0060.

#### **FSTORCK *int,lvm,svm***

Specifies how free storage availability should be monitored, meaning monitoring for the main Operations Manager server (OPMGRM1 by default and referred to as *lvm*) and each action processing server (OPMGRSn by default and referred to as *svm*). Free storage is checked every *int* minutes.

A message to the system operator is issued if the amount of contiguous free storage (in MB) drops below the size specified for the *lvm* or *svm*. If 0 is specified for *int*, *lvm*, or *svm*, the corresponding checks are not performed.

The maximum value for *lvm* and *svm* is 2047. Each value for *int*, *lvm*, and *svm* must be a four-digit number. The startup default is 0015,0016,0008.

#### **RULELIM *count,interval[,lmactn]***

Specifies the default LIMIT information for the DEFRULE command. The startup value is 1000,300. Refer to the LIMIT operand in the DEFRULE command, which is found in [“Operands” on page 110](#), for additional information.

#### **MMONLIM *count,interval[,lmactn]***

Specifies the default LIMIT information for the DEFMMON command. The startup value is 0,0. Refer to the LIMIT operand in the DEFMMON command, which is found in [“Operands” on page 101](#), for additional information.

#### **SCHDLIM *count,interval[,lmactn]***

Specifies the default LIMIT information for the DEFSCHD command. The startup value is 0,0. Refer to the LIMIT operand in the DEFSCHD command, which is found in [“Operands” on page 116](#), for additional information.

#### **SMONLIM *count,interval[,lmactn]***

Specifies the default LIMIT information for the DEFSMON command. The startup value is 0,0. Refer to the LIMIT operand in the DEFSMON command, which is found in [“Operands” on page 120](#), for additional information.

#### **EMONLIM *count,interval[,lmactn]***

Specifies the default LIMIT information for the DEFEMON command. The startup value is 1000,300. Refer to the LIMIT operand in the DEFEMON command, which is found in [“Operands” on page 91](#), for additional information.

**PMONLIM *count,interval[,lmactn]***

Specifies the default LIMIT information for the DEFPMON command. The startup value is 0,0. Refer to the LIMIT operand in the DEFPMON command, which is found in [“Operands” on page 106](#), for additional information.

**IMONLIM *count,interval[,lmactn]***

Specifies the default LIMIT information for the DEFIMON command. The startup value is 0,0. Refer to the LIMIT operand in the DEFIMON command, which is found in [“Operands” on page 95](#), for additional information.

**VCSCGRP *vcngrp***

Specifies the group name to control the method that VIEWCON uses to communicate with monitored users.

- If the monitored user is a member of the specified group, VIEWCON will use the guest's simulated hardware system console. This can be used to issue commands to z/OS® guests that have been configured to use the simulated hardware system console.
- If the monitored user is **not** a member of the specified group, VIEWCON will use the guest's virtual console. This is the method typically used to send commands to CMS and Linux guests.

The default group name is "\_GOMVCSC".

**SDFCMD *Y/N***

Specifies if the commands can be performed on the system data files using the VIEWSPPL command. These include the system files on the \*NSS, \*IMG, \*UCR, and \*NLS queues. The startup default is N.

**Related tasks**

[Step 5. \(Optional\) Configure an external security mechanism](#)

**Related reference**

[DEFACTN](#)

## Operand value lengths

[Table 26 on page 61](#) lists the operand value lengths for the DEFOPTN command.

<i>Table 26. DEFOPTN operand value lengths</i>	
<b>Operand</b>	<b>Length</b>
ACTNQCK	12
AUTHCMD	1
AUTHEXT	1 - 128
DSTADJ	1
ESM	1
ESMCLASS	1 - 8
ESMPREF	1 - 48
FSTORCK	12
HLDATTR	16
MIXCASE	1
SYMBOLS	1 - 16
EMONLIM	3 - 28
IMONLIM	3 - 28
MMONLIM	3 - 28

Table 26. DEFOPTN operand value lengths (continued)

Operand	Length
PMONLIM	3 - 28
SMONLIM	3 - 28
RULELIM	3 - 28
SCHDLIM	3 - 28
VCSCGRP	1-8
SDFCMD	1-1

## Example

An example is provided to help you understand how to use the DEFOPTN command and to show the typical results of this command.

### Example 1: Define options (DEFOPTN) in the startup configuration file for using Operations Manager.

In the example below, the options defined in the startup configuration file instruct Operations Manager to do the following:

- Treat mixed case input as not case sensitive.
- Define the search symbols as the forward slash and back slash (/ and \).
- Define the repeat symbol as the equal sign (=).
- Define the continuation symbols as the plus sign (+).
- Define line masking character for masking messages in the Operations Manager log as asterisk (\*).
- Set the DSTADJ operand to cause Operations Manager to trigger schedules (defined with DEFSCHD) based on the time zone change.

```
===== *
===== * Define options to the Operations Manager
===== *
===== DEFOPTN MIXCASE N SYMBOLS '/\=+*' DSTADJ N
===== *
```

Figure 21. DEFOPTN command

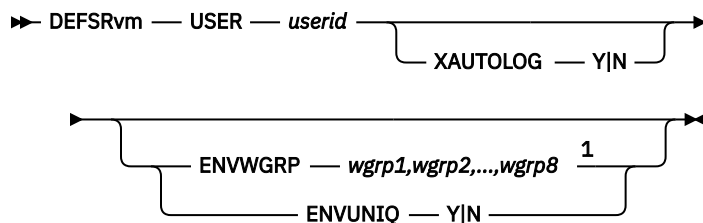
## Usage notes

1. One option must be specified.
2. Options not specified are not changed.

## Return codes

Table 27 on page 62 lists the return codes for the DEFOPTN command.

Table 27. DEFOPTN return codes	
Return code	Description
0	Finished correctly
-100	Error



Notes:

<sup>1</sup> Specify up to eight (8) values with each value separated by a comma.

Figure 22. DEFSRVM command syntax

## Authorization

The user issuing this command must be authorized for control commands.

## Purpose

Use the DEFSRVM command to define an action processing service machine. The service machine is available to process actions assigned to it by OPMGRM1, or the server running GOMMAIN.

## Operands

### USER *userid*

Specifies the VM user ID of the service machine that you want to define.

### XAUTOLOG Y|N

Specifies if at startup OPMGRM1 is to issue the CP XAUTOLOG command to auto-log the action processing server. This operand is optional, but highly recommended to use XAUTOLOG Y. The default is XAUTOLOG N.

### ENVUNIQ Y|N

Specifies if the service machine only processes an ACTION (specified by DEFACTN) where the ENV operand value matches the *userid* of the service machine. ENVUNIQ Y prevents the service machine from processing an ACTION where ENV SVM is specified. The default is ENVUNIQ N.

### ENVWGRP *wgrp1,wgrp2,...,wgrp8*

Specifies that the service machine only processes an ACTION where the ENV operand value matches one of the work groups. A service machine may participate in up to eight (8) work groups. This prevents the service machine from processing an ACTION where ENV SVM is specified.

### Related reference

[DEFACTN](#)

## Operand value lengths

Table 28 on page 63 lists the operand value lengths for the DEFSRVM command.

Table 28. DEFSRVM operand value lengths	
Operand	Length
ENVWGRP	1-72
ENVUNIQ	1
USER	1-8
XAUTOLOG	1

# Example

An example is provided to help you understand how to use the DEFSRVM command and to show the typical results of this command.

## Example 1: Define four action processing servers (OPMGRS1, OPMGRS2, OPMGRS3 and OPMGRS4) and automatically log them on to run disconnected when OPMGRM1 is started.

```
===== * Define action processing servers to Operations Manager
===== *
===== DEFSRVM USER OPMGRS1 XAUTOLOG Y
===== DEFSRVM USER OPMGRS2 XAUTOLOG Y
=====>

===== DEFSRVM USER OPMGRS3 XAUTOLOG Y
===== DEFSRVM USER OPMGRS4 XAUTOLOG Y
```

Figure 23. DEFSRVM commands in config file

# Usage notes

- 1. If XAUTOLOG (Y) is specified, the VM user ID running GOMMAIN must have PRIV A or B to operate correctly. For more information on privilege classes, refer to [Chapter 2, “Configuring Operations Manager,”](#) on page 5.
- 2. ENVUNIQ and ENVWRGP are mutually exclusive.

# Return codes

Table 29 on page 64 lists the return codes for the DEFSRVM command.

Table 29. DEFSRVM return codes	
Return code	Description
0	Finished correctly
-100	Error

# DEFVIEW

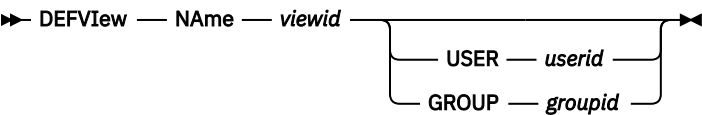


Figure 24. DEFVIEW command syntax

# Authorization

The user issuing this command must be authorized for control commands.

# Purpose

Issue the DEFVIEW command to define a combined view of multiple users' consoles. In addition to maintaining the user's console in its own view, this command adds users to a combined view. Issue the VIEWCON command to view the output.

# Operands

**NAme** *viewid*  
Specifies a name for the view.



**USER *userid***

Specifies a user ID or a wildcard pattern to match against a user's ID. Output collected for the user ID, or any user name matching the pattern, is added to the view.

**GROUP *groupid***

Specifies a group name. Output collected for any user who is a member of the group is added to the view.

**Related concepts**

[Wildcard pattern matching](#)

Wildcard pattern matching is used by several of the commands. Wildcards allow you to search for and process characters that are similar.

**Related reference**

[DEFGROUP](#)

[VIEWCON](#)

## Operand value lengths

Table 30 on page 65 lists the operand value lengths for the DEFVIEW command.

Table 30. DEFVIEW operand value lengths	
Operand	Length
NAME	1-8
USER	1-8
GROUP	1-8

## Examples

This section provides examples on how to use the DEFVIEW command.

### Example 1: Define a combined view named **SAMPLE** that includes consoles from monitored users **MAINT640** and **TS3163**.

```
GOMCMD OPMGRM1 DEFVIEW NAME SAMPLE USER MAINT640
Ready; T=0.01/0.01 06:05:00
GOMCMD OPMGRM1 DEFVIEW NAME SAMPLE USER TS3163
Ready; T=0.01/0.01 06:05:10
```

Figure 25. DEFVIEW command with two users

After the view is defined, use the **VIEWCON** command to view the output.

```
VIEWCON USER SAMPLE
```

Figure 26. VIEWCON command

### Example 2: Members (**TS3163** and **MAINT64**) of a defined group (**ROCKET1**) are added to a defined view (**ROCVIEW**).

```
GOMCMD OPMGRM1 DEFGROUP NAME ROCKET1 MEMBER TS3163
Ready; T=0.01/0.01 01:45:25
GOMCMD OPMGRM1 DEFGROUP NAME ROCKET1 MEMBER MAINT640
Ready; T=0.01/0.01 01:45:34
GOMCMD OPMGRM1 DEFVIEW NAME ROCVIEW GROUP ROCKET1
Ready; T=0.01/0.01 01:52:07
```

Figure 27. DEFGROUP commands combined with DEFVIEW

# Usage notes

1. Either USER or GROUP must be specified.
2. USER and GROUP are mutually exclusive.
3. If you want to add multiple user IDs to a combined view, you can use a wildcard pattern and/or specify one DEFVIEW command for each user ID. For example, to add OPERATOR and all user IDs starting with "USER" into the combined view named MYVIEW, specify these DEFVIEW commands.

```
DEFVIEW NAME MYVIEW USER OPERATOR
DEFVIEW NAME MYVIEW USER USER*
```

4. The output from a user's console may be added to multiple combined views. The output from a combined view cannot be included in other combined views.
5. Issue the VIEWCON command to view the output from a combined view. To issue a command to a user's console defined in the combined view, you must put the target ID in front of the command. For example, USER1 *command*.
6. See [Appendix B, "Wildcard pattern matching," on page 181](#) for an explanation of wildcard pattern matching.

# Return codes

Table 31 on page 66 lists the return codes for the DEFVIEW command.

Table 31. DEFVIEW return codes	
Return code	Description
0	Finished correctly
-100	Error

# DELACTION

➡ DELACTION — TOKEN — *token* ➡

Figure 28. DELACTION command syntax

# Authorization

The user issuing this command must be authorized for control commands.

# Purpose

Use the DELACTION command to delete an entry from a list of actions that are currently queued for execution by OPMGRM1 or an action processing server.

# Operands

**TOKEN** *token*

Specifies the token of the processing entry. The token is a 16-byte value displayed by the STATUS  
DETAIL ACTIONQ output.

# Operand value lengths

The table below lists the operand value lengths for the DELACTION command.

Table 32. DELACTQ operand value lengths	
Operand	Length
TOKEN	16

## Examples

An example is provided to help you understand how to use the DELACTQ command and to show the typical results of this command.

### Example 1: Use the STATUS command to see the list of actions in the queue and use the DELACTQ command to delete an action from the queue.

The example screen below shows the full list of actions in queue for execution. Notice that there is an action named TEST2 and another named TEST3 on this list.

```
GOMCMD OPMGRM1 STATUS DETAIL ACTIONQ
09/03/19 02:26:41 GOMCMD0945I ACTION TEST3 SCHEDULED FOR PROCESSING (_GOMSCHD/TESTSCH3/
USERID /ZVM6KT02/D6A6487805D1A241)
09/03/19 02:26:41 GOMCMD2904I ELIGIBLE FOR PROCESSING AT 2019/08/30-06:51:39
09/03/19 02:26:41 GOMCMD0945I ACTION TEST2 SCHEDULED FOR PROCESSING (_GOMSCHD/
TESTSCH2/SVM /ZVM6KT02/D6A6481DD55D8742)
09/03/19 02:26:41 GOMCMD2904I ELIGIBLE FOR PROCESSING AT 2019/08/30-06:50:05
09/03/19 02:26:41 GOMCMD2999I STATUS DETAIL COMPLETE
Ready; T=0.01/0.01 02:26:41
GOMCMD OPMGRM1 DELACTQ TOKEN D6A6487805D1A241
Ready; T=0.01/0.01 02:30:49
```

Figure 29. List of actions before the DELACTQ command is issued

```
GOMCMD OPMGRM1 DELACTQ TOKEN D6A6487805D1A241
Ready; T=0.01/0.01 02:30:49
```

Figure 30. DELACTQ command issued

After issuing the DELACTQ command, you can see on the revised status screen that TEST2 is in the queue, but TEST3 has been removed.

```
GOMCMD OPMGRM1 STATUS DETAIL ACTIONQ
09/03/19 02:26:41 GOMCMD0945I ACTION TEST2 SCHEDULED FOR PROCESSING (_GOMSCHD/
TESTSCH2/SVM /ZVM6KT02/D6A6481DD55D8742)
09/03/19 02:26:41 GOMCMD2904I ELIGIBLE FOR PROCESSING AT 2019/08/30-06:50:05
09/03/19 02:26:41 GOMCMD2999I STATUS DETAIL COMPLETE
Ready; T=0.01/0.01 02:26:41
```

Figure 31. List of actions after the DELACTQ command is issued

## Usage notes

See the STATUS DETAIL ACTIONQ output for the processing list.

## Return codes

The table below lists the return codes for the DELACTQ command.

Table 33. DELACTQ return codes	
Return code	Description
0	Finished correctly
-100	Error

# DELDSPS

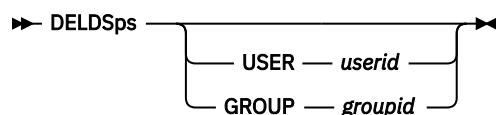


Figure 32. DELDSPS command syntax

## Authorization

The user issuing this command must be authorized for control commands.

## Purpose

Issue the DELDSPS command to delete the size definition (the template used by the DEFDSPS command) for a previously defined data space. The default size is used instead. This size definition specifies the amount of virtual storage (or memory) used to capture the data for a monitored user's console or combined view. The DELDSPS command deletes a template that was used by the DEFDSPS command when defining the size of a data space. DELDSPS does not delete or modify any actual z/VM data spaces.

## Operands

### USER *userid*

Specifies a user name, a combined view name, or a wildcard pattern to delete.

### GROUP *groupid*

Specifies a group name to delete.

### Related reference

[DEFDSPS](#)

[DEFGROUP](#)

## Operand value lengths

Table 34 on page 68 lists the operand value lengths for the DELDSPS command.

Table 34. DELDSPS operand value lengths	
Operand	Length
USER	1-8
GROUP	1-8

## Examples

A few examples are provided to help you understand how to use the DELDSPS command to delete the size specification from a specific user or group.

**Example 1: Delete the storage size specification that was previously assigned to the console data for the monitored user MAINT640. The storage size for console data for MAINT640 will revert to the default value.**

```
GGOMCMD OPMGRM1 DELDSPS USER MAINT640
Ready; T=0.01/0.01 05:15:54
```

Figure 33. DELDSPS command issued for user MAINT640

**Example 2: Delete the storage size specification that was previously assigned to the console data for the monitored group ROCKET1. The storage size for console data for ROCKET1 will revert to the default value.**

```
GOMCMD OPMGRM1 DELDSPS GROUP ROCKET1
Ready; T=0.01/0.01 02:22:45
```

Figure 34. DELDSPS command issued for group ROCKET1

## Usage notes

1. Either USER or GROUP must be specified.
2. USER and GROUP are mutually exclusive.
3. The value of the USER or GROUP operand must completely match a previously defined value, including wildcard characters.
4. See Appendix B, “Wildcard pattern matching,” on page 181 for an explanation of wildcard pattern matching.

## Return codes

Table 35 on page 69 lists the return codes for the DELDSPS command.

Table 35. DELDSPS return codes	
Return code	Description
0	Finished correctly
-100	Error

## DELIPCS

➡ DELIPcs — NAmE — ipcsid ➡

Figure 35. DELIPCS command syntax

## Authority

The user issuing this command must be authorized for control commands.

## Purpose

Use the DELIPCS command to delete an Internet Protocol Communication Session (IPCS).

## Operands

**NAmE ipcsid**  
Specifies the name of the IP session.

## Operand value lengths

The table below lists the operand value lengths for the DELIPCS command.

Table 36. DELIPCS operand value lengths	
Operand	Length
NAME	1-8

# Example

An example is provided to help you understand how to use the DELIPCS command, and shows the typical results of this command.

## Example 1: Use the DELIPCS command to delete the Internet Protocol Communication Session named SAMPLE2.

```
GOMCMD OPMGRM1 DELIPCS NAME SAMPLE2
Ready; T=0.01/0.01 06:02:03
```

Figure 36. DELIPCS command

# Usage notes

None

# DELSRVM

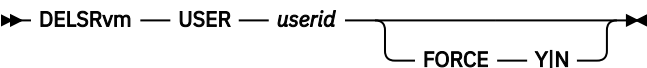


Figure 37. DELSRVM command syntax

# Authorization

The user issuing this command must be authorized for control commands.

# Purpose

Issue the DELSRVM command to delete an action processing service machine from the current Operations Manager configuration.

## Related reference

[DEFSRVM](#)

# Operands

## USER *userid*

Specifies the VM user ID of the action processing server (service machine) running GOMSVM that you want to delete.

## FORCE Y|N

Specifies if the deletion of the service machine is forced to ensure that the deletion occurs. Forced deletion also issues a CP FORCE command to log off the user ID. The default is FORCE N.

# Operand value lengths

Table 37 on page 70 lists the operand value lengths for the DELSRVM command.

Table 37. DELSRVM operand value lengths	
Operand	Length
FORCE	1
USER	1-8

## Example

An example is provided to help you understand how to use the DELSRVM command, and shows the typical results of this command.

### Example 1: Use the DELSRVM command to delete the action processing server OPMGRS4.

```
GOMCMD OPMGRM1 DELSRVM USER OPMGRS4
Ready; T=0.01/0.01 06:02:03
```

Figure 38. DELSRVM command

## Usage notes

1. If **FORCE Y** is specified, the VM user ID running GOMMAIN must have **PRIV A** to operate correctly. For more information on privilege classes, refer to [Chapter 2, “Configuring Operations Manager,” on page 5](#).
2. If the action processing server (service machine) is busy, the definition will not be deleted unless the **FORCE Y** operand is specified.
3. If **FORCE N** is specified and the action processing server is busy, a message is issued and the command is ignored. The action processing server is not deleted even when it becomes idle.

## Return codes

Table 38 on page 71 lists the return codes for the DELSRVM command.

Table 38. DELSRVM return codes	
Return code	Description
0	Finished correctly
-100	Error

## DELVIEW

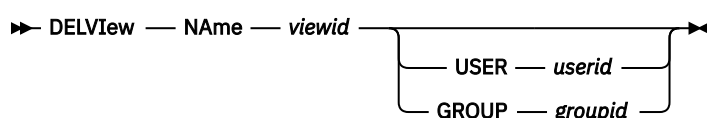


Figure 39. DELVIEW command syntax

## Authorization

The user issuing this command must be authorized for control commands.

## Purpose

Use the DELVIEW command to delete the definition of a combined view. This command deletes the user(s) or a group from a combined view. The output for the user(s) or group is no longer added to the named view. This command does not delete any previously combined output.

## Operands

**NAME** *viewid*

Specifies the name of the view to delete.

**USER *userid***

Specifies a user name or a wildcard pattern to delete.

**GROUP *groupid***

Specifies a group name to delete.

**Related reference**

[DEFGROUP](#)

[DEFVIEW](#)

## Operand value lengths

Table 39 on page 72 lists the operand value lengths for the DELVIEW command.

Table 39. DELVIEW operand value lengths	
Operand	Length
NAME	1-8
USER	1-8
GROUP	1-8

## Examples

The examples below show how to use the DELVIEW command to delete a user or a group from the combined view.

**Example 1: Delete a specific user, MAINT640, from a combined view, SAMPLE.**

```
GOMCMD OPMGRM1 DELVIEW NAME SAMPLE USER MAINT640
Ready; T=0.01/0.01 06:05:00
```

Figure 40. DELVIEW command to delete a user

After deletion, the output for user MAINT640 will no longer be added to the SAMPLE view. However, this command does not delete any previously combined output.

**Example 2: A specific group of users, ROCKET1, is deleted from a view, ROCVIEW.**

```
GOMCMD OPMGRM1 DELVIEW NAME ROCVIEW GROUP ROCKET1
Ready; T=0.01/0.01 02:16:02
```

Figure 41. DELVIEW command to delete a group

After deletion, the group's output will no longer be added to the ROCVIEW view. However, this command does not delete any previously combined output.

## Usage notes

1. Either USER or GROUP must be specified.
2. USER and GROUP are mutually exclusive.
3. The values of the NAME and USER/GROUP operands must completely match previously defined values.
4. Only the unique combination of NAME and USER/GROUP are removed. Other NAME and USER/GROUP combinations are not removed.
5. See [Appendix B, “Wildcard pattern matching,” on page 181](#) for an explanation of wildcard pattern matching.



## Return codes

Table 40 on page 73 lists the return codes for the DELVIEW command.

Table 40. DELVIEW return codes	
Return code	Description
0	Finished correctly
-100	Error

## QUIT

➡ QUIT ➡

Figure 42. QUIT command syntax

## Authorization

The user issuing this command must be authorized for control commands.

## Purpose

Issue the QUIT command to stop Operations Manager (GOMMAIN). In a default installation, this is the user ID: OPMGRM1. There are no operands and none are allowed.

## Example

An example is provided to help you understand how to use the QUIT command. Note that QUIT and STOP function in the same way. Issue either one of these commands to stop Operations Manager processing.

**Example 1: Use the QUIT command to halt any processing currently in progress on OPMGRM1 .**

```
GOMCMD OPMGRM1 QUIT
```

Figure 43. QUIT command issued

## Usage notes

1. QUIT and STOP are equivalent commands. Issue one of these commands to stop Operations Manager (GOMMAIN) processing.

## Return codes

Table 41 on page 73 lists the return codes for the QUIT command.

Table 41. QUIT return codes	
Return code	Description
0	Finished correctly
-100	Error

# REVOKE

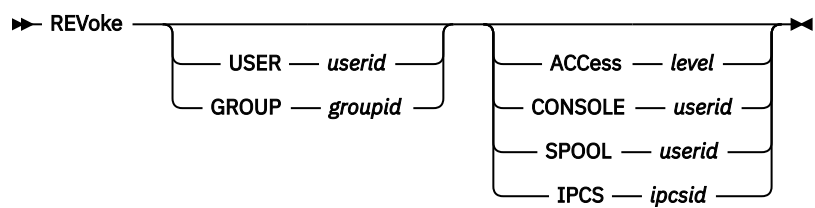


Figure 44. REVOKE command syntax

## Authorization

The user issuing this command must be authorized for control commands.

## Purpose

Issue the REVOKE command to remove authorization from a VM user or group to issue certain Operations Manager commands, view the consoles or spooled output of certain users, and interact with IP sessions . If you are using an ESM, do not issue the REVOKE command.

## Operands

### **USER** *userid*

Identifies the user whose authority is being revoked.

### **GROUP** *groupid*

Identifies the group whose authorities are being revoked.

### **ACCESS** *level*

Specifies the level of Operations Manager commands being removed from the user or group. The *level* may be CONTROL or CONFIG.

### **CONSOLE** *userid*

Specifies that the user's or group's authority to view and interact with the console belonging to *userid*, or to a combined view named *userid*, is being revoked.

### **SPOOL** *userid*

Specifies that the user's or group's authority to view the spooled output belonging to *userid* is being revoked.

### **IPCS** *ipcsid*

Specifies that the user's or group's authority to interact with the IP session, defined as *ipcsid*, is being revoked.

### **Related tasks**

[Step 5. \(Optional\) Configure an external security mechanism](#)

### **Related reference**

[AUTH](#)

[DEFGROUP](#)

## Operand value lengths

Table 42 on page 74 lists the operand value lengths for the REVOKE command.

Table 42. REVOKE operand value lengths	
Operand	Length
ACCESS	6-7

Table 42. REVOKE operand value lengths (continued)	
Operand	Length
CMDIPCS	1-8
CONSOLE	1-8
GROUP	1-8
SP00L	1-8
USER	1-8

## Example

An example is provided to help you understand how to use the REVOKE command, and the typical results of this command.

### Example 1: The REVOKE command removes authority from a VM user, TS3163, to prevent the issuing of control commands.

```
GOMCMD OPMGRM1 REVOKE USER TS3163 ACCESS CONTROL
Ready; T=0.01/0.01 01:24:45
```

Figure 45. REVOKE command issued

Now when user TS3163 tries to issue control commands, like AUTH, an error message is generated stating that user TS3163 does not have the authority to issue this AUTH command. An example is shown below.

```
GOMCMD OPMGRM1 AUTH USER TS3163 ACCESS CONTROL
09/13/19 01:25:44 GOMCMD0203E TS3163 IS NOT AUTHORIZED FOR AUTH
Ready(-0100); T=0.01/0.01 01:25:44
```

Figure 46. "Not authorized" message

## Usage notes

1. Either USER or GROUP must be specified.
2. The USER operand and the GROUP operand are mutually exclusive.
3. Each REVOKE command may have only one operand (ACCESS, CONSOLE, SP00L, or IPCS).
4. The *userid* value of the CONSOLE operand must match the value specified in the AUTH command, including wildcard characters.
5. The *userid* value of the SP00L operand must match the value specified in the AUTH command, including wildcard characters.
6. The *ipcsid* value of the IPCS operand must match the value specified in the AUTH command, including wildcard characters.

## Return codes

Table 43 on page 75 lists the return codes for the REVOKE command.

Table 43. REVOKE return codes	
Return code	Description
0	Finished correctly
-100	Error

# STATUS

---

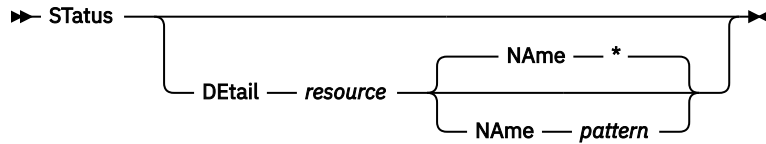


Figure 47. STATUS command syntax

## Authorization

The user issuing this command must be authorized for configuration or control commands.

## Purpose

Issue the STATUS command to show the current status of Operations Manager (GOMMAIN) processing.

- If the DETAIL operand is *not* present, a summary is provided as output.
- If the DETAIL operand is present, additional information about each specified resource is provided as output.

## Operands

### DEtail *resource*

Requests additional information for the specified resource definitions.

The valid resources are:

- ACTION — Displays the status of the defined actions.
- ACTIONQ — Displays the actions queued for execution.
- AUTH — Displays the status of authorized users.
- CONSOLE — Displays the status of the recording consoles.
- DATASP — Displays the status of the defined data space template sizes.
- DATE — Displays the status of the defined holidays.
- EVENT — Displays the status of the defined event monitors.
- GROUP — Displays the status of the defined groups.
- IDLE — Displays the status of the defined idle monitors.
- IPCS — Displays the status of the IP sessions.
- IUCV — Displays the status of the active IUCV connections.
- MACHINE — Displays the status of the monitored virtual machines.
- OPTION — Displays the status of the current options settings.
- PAGE — Displays the status of the defined page space monitors.
- RULE — Displays the status of the defined rules.
- SCHEDULE — Displays the status of the defined schedules.
- SERVICE — Displays the status of the defined action processing servers.
- SPOOL — Displays the status of the defined spool usage monitors.
- SPOOLUSR — Displays the top ten largest spool files and the top ten users with the most spool files.
- VIEW — Displays the status of the combined views.

### NAme *pattern*

Lists information for resource names which match the pattern. Wildcard patterns are allowed. The wildcard pattern rules are explained in [Appendix B, “Wildcard pattern matching,” on page 181](#).

## Related reference

[DEFRULE](#)

## Operand value lengths

Table 44 on page 77 lists the operand value lengths for the STATUS command.

Table 44. STATUS operand value lengths	
Operand	Length
DETAIL	1-8
NAME	1-8

## Examples

### Example 1: Check the current status of Operations Manager (GOMMAIN) processing.

The following examples show how to use the STATUS command. An explanation of the command and the results it generates are shown for each example.

```
GOMCMD OPMGRM1 STATUS
09/09/19 00:27:33 GOMCMD0899I OPERATIONS MANAGER V1R6 BUILT 2018/08/29 16:45
09/09/19 00:27:33 GOMCMD0900I 2 RULE(S) DEFINED
09/09/19 00:27:33 GOMCMD0902I 2 MACHINE MONITOR(S) DEFINED
09/09/19 00:27:33 GOMCMD0903I 4 SCHEDULE(S) DEFINED
09/09/19 00:27:33 GOMCMD0904I 5 ACTION(S) DEFINED
09/09/19 00:27:33 GOMCMD0905I 0 ACTION(S) SCHEDULED FOR PROCESSING
09/09/19 00:27:33 GOMCMD0906I 4 GROUP(S) DEFINED
09/09/19 00:27:33 GOMCMD0907I 0 DATES(S) DEFINED
09/09/19 00:27:33 GOMCMD0908I 0 SPOOL MONITOR(S) DEFINED
09/09/19 00:27:33 GOMCMD0909I 0 EVENT MONITOR(S) DEFINED
09/09/19 00:27:33 GOMCMD0910I 0 PAGE MONITOR(S) DEFINED
09/09/19 00:27:33 GOMCMD0911I 0 IDLE MONITOR(S) DEFINED
09/09/19 00:27:33 GOMCMD0920I 5 SERVICE MACHINE(S) DEFINED
09/09/19 00:27:33 GOMCMD0921I 4 USER(S)/GROUP(S) AUTHORIZED
09/09/19 00:27:33 GOMCMD0922I 4 IUCV CONNECTION(S)
09/09/19 00:27:33 GOMCMD0923I 0 DATA SPACE SIZE DEFINITION(S)
09/09/19 00:27:33 GOMCMD0924I 3 COMBINED VIEW(S)
09/09/19 00:27:33 GOMCMD0925I 13 USER CONSOLE(S) RECORDING
09/09/19 00:27:33 GOMCMD0926I 0 IP SESSION(S) DEFINED
Ready; T=0.01/0.01 00:27:33
```

Figure 48. Summary output using STATUS command without operands

**Example 2: Use STATUS with the DETAIL operand to list each user and the type(s) of authority currently assigned to the user(s).**

```
OMCMD OPMGRM1 STATUS DETAIL AUTH
09/09/19 00:32:06 GOMCMD2928I USER TS3163A AUTHORIZED FOR:
09/09/19 00:32:06 GOMCMD2995I CONTROL COMMANDS
09/09/19 00:32:06 GOMCMD2994I CONFIGURATION COMMANDS
09/09/19 00:32:06 GOMCMD2993I CONSOLE FOR *
09/09/19 00:32:06 GOMCMD2992I SPOOL FOR *
09/09/19 00:32:06 GOMCMD2928I USER TS3163 AUTHORIZED FOR:
09/09/19 00:32:06 GOMCMD2995I CONTROL COMMANDS
09/09/19 00:32:06 GOMCMD2994I CONFIGURATION COMMANDS
09/09/19 00:32:06 GOMCMD2993I CONSOLE FOR *
09/09/19 00:32:06 GOMCMD2992I SPOOL FOR *
09/09/19 00:32:06 GOMCMD2928I USER MAINT640 AUTHORIZED FOR:
09/09/19 00:32:06 GOMCMD2995I CONTROL COMMANDS
09/09/19 00:32:06 GOMCMD2994I CONFIGURATION COMMANDS
09/09/19 00:32:06 GOMCMD2993I CONSOLE FOR *
09/09/19 00:32:06 GOMCMD2992I SPOOL FOR *
09/09/19 00:32:06 GOMCMD2927I GROUP SYSADM AUTHORIZED FOR:
09/09/19 00:32:06 GOMCMD2995I CONTROL COMMANDS
09/09/19 00:32:06 GOMCMD2994I CONFIGURATION COMMANDS
09/09/19 00:32:06 GOMCMD2993I CONSOLE FOR *
09/09/19 00:32:06 GOMCMD2992I SPOOL FOR *
09/09/19 00:32:06 GOMCMD2929I INTERACTION FOR IP SESSION *
09/09/19 00:32:06 GOMCMD2999I STATUS DETAIL COMPLETE
Ready; T=0.01/0.01 00:33:09
```

Figure 49. STATUS command with DETAIL AUTH operand

**Example 3: Check the status of the recording consoles, such as the size and the number of messages.**

```
GOMCMD OPMGRM1 STATUS DETAIL CONSOLE
09/09/19 00:59:42 GOMCMD0965I CONSOLE SAMPLE2-ZVM6KT02 ASIT 00EDA5C0000000051 A
LET 0100000F SIZE 1M MSGS 1
09/09/19 00:59:42 GOMCMD0965I CONSOLE _GOMSCHD-ZVM6KT02 ASIT 00EDA9C0000000003 A
LET 0100000E SIZE 1M MSGS 33
09/09/19 00:59:42 GOMCMD0965I CONSOLE TMDMM-ZVM6KT02 ASIT 00EDA98000000000C A
LET 0100000D SIZE 1M MSGS 5
09/09/19 00:59:42 GOMCMD0965I CONSOLE TMLM1-ZVM6KT02 ASIT 00EDA8C0000000015 A
LET 0100000C SIZE 1M MSGS 5
09/09/19 00:59:42 GOMCMD0965I CONSOLE TMTMM-ZVM6KT02 ASIT 00EDA900000000069 A
LET 0100000B SIZE 1M MSGS 10
09/09/19 00:59:42 GOMCMD0965I CONSOLE KNAYAK-ZVM6KT02 ASIT 00EDA88000000005E A
LET 0100000A SIZE 1M MSGS 34
09/09/19 00:59:42 GOMCMD0965I CONSOLE SAMPLE-ZVM6KT02 ASIT 00EDA60000000005F A
LET 01000009 SIZE 1M MSGS 50
09/09/19 00:59:42 GOMCMD0965I CONSOLE _GOMMMON-ZVM6KT02 ASIT 00EDA7000000010E A
LET 01000008 SIZE 1M MSGS 7896
09/09/19 00:59:42 GOMCMD0965I CONSOLE OPERATOR-ZVM6KT02 ASIT 00EDAA00000000087 A
LET 01000007 SIZE 1M MSGS 1
09/09/19 00:59:42 GOMCMD0965I CONSOLE 5697J10F-ZVM6KT02 ASIT 00EDA580000001DB A
LET 01000006 SIZE 1M MSGS 1
09/09/19 00:59:42 GOMCMD0965I CONSOLE TS3163-ZVM6KT02 ASIT 00EDA940000000073
LET 01000005 SIZE 1M MSGS 40
09/09/19 00:59:42 GOMCMD0965I CONSOLE MAINT640-ZVM6KT02 ASIT 00EDA8000000000CC A
LET 01000004 SIZE 1M MSGS 39
09/09/19 00:59:42 GOMCMD0965I CONSOLE RACFVM-ZVM6KT02 ASIT 00EDA7C00000000D8 A
LET 01000003 SIZE 1M MSGS 4
09/09/19 00:59:42 GOMCMD2999I STATUS DETAIL COMPLETE
Ready; T=0.01/0.01 01:00:02
```

Figure 50. STATUS command with DETAIL CONSOLE operands

#### Example 4: Check the status and the details of monitored virtual machines.

```
STATUS DETAIL (MACHINE)
2019-05-26 11:46:16 GOMCMD0942I MACHINE MONITOR MON1 TRIGGERED 0 TIME(S)
2019-05-26 11:46:16 GOMCMD2977I ACTION MACHAC
2019-05-26 11:46:16 GOMCMD2976I DAYS ALL
2019-05-26 11:46:16 GOMCMD2975I HOURS ALL
2019-05-26 11:46:16 GOMCMD2973I USER RSCS (RSCS - DSC)
2019-05-26 11:46:16 GOMCMD2972I INTERVAL 1 MINUTE(S)
2019-05-26 11:46:16 GOMCMD2971I DELAY 0 MINUTE(S)
2019-05-26 11:46:16 GOMCMD2970I RETRY INTERVAL 1 MINUTE(S) RETRY COUNT 0
2019-05-26 11:46:16 GOMCMD2968I NEXT CHECK 2019/05/26 11:46
2019-05-26 11:46:16 GOMCMD2999I STATUS DETAIL COMPLETE
```

Figure 51. Details about monitored virtual machines

#### Example 5: Check the status and details of spool usage monitors.

```
STATUS DETAIL (SPOOL)
2019-05-26 11:48:49 GOMCMD0948I SPOOL MONITOR WARN TRIGGERED 0 TIME(S)
2019-05-26 11:48:49 GOMCMD2977I ACTION ACTNSP
2019-05-26 11:48:49 GOMCMD2976I DAYS ALL
2019-05-26 11:48:49 GOMCMD2975I HOURS ALL
2019-05-26 11:48:49 GOMCMD2967I USAGE LOW 80 HIGH 89
2019-05-26 11:48:49 GOMCMD2972I INTERVAL 1 MINUTE(S)
2019-05-26 11:48:49 GOMCMD2968I NEXT CHECK 2019/05/26 11:48
2019-05-26 11:48:49 GOMCMD0948I SPOOL MONITOR CRIT TRIGGERED 0 TIME(S)
2019-05-26 11:48:49 GOMCMD2977I ACTION ACTNSP
2019-05-26 11:48:49 GOMCMD2976I DAYS ALL
2019-05-26 11:48:49 GOMCMD2975I HOURS ALL
2019-05-26 11:48:49 GOMCMD2967I USAGE LOW 90 HIGH 100
2019-05-26 11:48:49 GOMCMD2972I INTERVAL 1 MINUTE(S)
2019-05-26 11:48:49 GOMCMD2968I NEXT CHECK 2019/05/26 11:48
2019-05-26 11:48:49 GOMCMD2999I STATUS DETAIL COMPLETE
```

Figure 52. Details about all spool usage monitors

#### Example 6: Check the status and details of users' spool files.

```
status detail spoolusr
2019-05-26 11:51:11 GOMCMD0970I USER OPERATNS SPOOL FILE ID 0175 IS USING 8401 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0970I USER RWHITE2 SPOOL FILE ID 0445 IS USING 7192 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0970I USER RWHITE2 SPOOL FILE ID 0446 IS USING 7192 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0970I USER SAMARAK SPOOL FILE ID 0070 IS USING 6167 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0970I USER RWHITE2 SPOOL FILE ID 0448 IS USING 6030 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0970I USER SAMARAK SPOOL FILE ID 0021 IS USING 5051 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0970I USER SMARAK SPOOL FILE ID 0003 IS USING 4226 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0970I USER RWHITE2 SPOOL FILE ID 0450 IS USING 3483 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0970I USER PERFSVM SPOOL FILE ID 0383 IS USING 2986 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0970I USER PERFSVM SPOOL FILE ID 0380 IS USING 2985 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0971I USER BKADMIN HAS 1673 SPOOL FILES USING 59605 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0971I USER DMARTIN HAS 609 SPOOL FILES USING 23093 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0971I USER MAINT HAS 351 SPOOL FILES USING 1713 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0971I USER SAMARAK HAS 229 SPOOL FILES USING 88504 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0971I USER TCPMAINT HAS 218 SPOOL FILES USING 2873 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0971I USER MAINT640 HAS 217 SPOOL FILES USING 994 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0971I USER CSHOWA HAS 215 SPOOL FILES USING 879 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0971I USER CSHOW2 HAS 207 SPOOL FILES USING 1262 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0971I USER OPERATOR HAS 207 SPOOL FILES USING 10618 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0971I USER MDMARA HAS 199 SPOOL FILES USING 4599 SPOOL BLOCKS
```

Figure 53. Details about users' spool files

## Usage notes

None

## Return codes

Table 45 on page 80 lists the return codes for the STATUS command.

Table 45. STATUS return codes	
Return code	Description
0	Finished correctly
-100	Error

## STOP

➤ STOP ➤

Figure 54. STOP command syntax

### Authorization

The user issuing this command must be authorized for control commands.

### Purpose

Issue the STOP command to stop Operations Manager (GOMMAIN). In a default installation, this is the user ID OPMGRM1. There are no operands and none are allowed.

### Example

An example is provided to help you understand how to use the STOP command. Note that STOP and QUIT function in the same way. Issue either one of these commands to stop Operations Manager processing.

#### Example 1: Use the STOP command to halt any processing.

```
GOMCMD OPMGRM1 STOP
```

Figure 55. STOP command issued

The Virtual Machine must be restarted to continue working with Operations Manager.

### Usage note

QUIT and STOP are equivalent commands. Issue one of these commands to stop GOMMAIN processing.

### Return codes

Table 46 on page 80 lists the return codes for the STOP command.

Table 46. STOP return codes	
Return code	Description
0	Finished correctly
-100	Error



---

## Chapter 8. Syntax for Configuration Commands

This chapter describes the Operations Manager configuration commands and their syntax.

### Related tasks

[Step 5. \(Optional\) Configure an external security mechanism](#)

## Purpose of configuration commands

---

Configuration commands allow you to define specific conditions you want Operations Manager to process and what action is taken when that condition occurs. These conditions include:

- rules
- schedules
- machine monitors
- spool monitors
- page space monitors
- event monitors

You can type a configuration command on the Operations Manager console or issue it using the GOMCMD command interface. Both methods execute the command immediately. You can also put configuration commands into a file so that each time you start Operations Manager, the same rules, schedules, monitors, and actions are in effect and you do not have to re-enter them.

Some of these configuration commands are also valid as control commands.

### Related reference

[Creating a startup configuration file](#)

[Syntax for control commands](#)

## Authorization for configuration commands

---

You must use an ESM or the AUTH command to give a specific user the authority to issue configuration commands. To give users access to specific configuration commands, an ESM is required.

### Related tasks

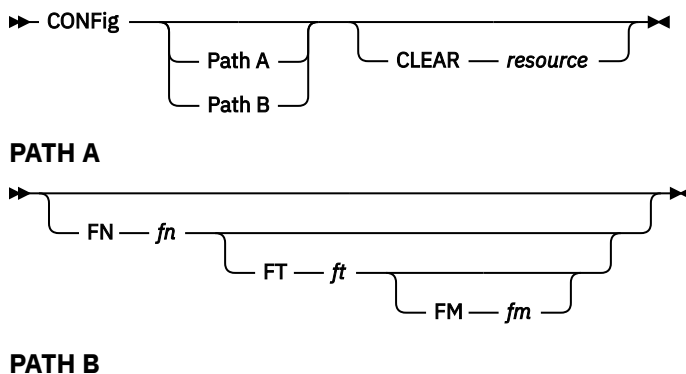
[Step 5. \(Optional\) Configure an external security mechanism](#)

### Related reference

[AUTH](#)

## CONFIG

---



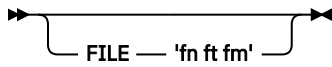


Figure 56. CONFIG command syntax

## Authorization

The user issuing this command must be authorized for configuration or control commands. If CONFIG is entered via the GOMCMD interface, SMSG interface, or VIEWLOG command line, each command in the specified "config" file is checked for authorization.

## Purpose

Use the CONFIG command to process the contents of a file containing Operations Manager commands.

## Operands

### **FN** *fn*

Specifies the file name of the file to be processed. The default is MAS.

### **FT** *ft*

Specifies the file type of the file to be processed. The default is COMMANDS.

### **FM** *fm*

Specifies the file mode of the file to be processed. The default is A1. This is the file mode on OPMGRM1.

### **FILE** '*fn ft fm*'

Specifies the file name, file type, and file mode to be processed. The default is MAS COMMANDS A1.

### **CLEAR** *resource*

Indicates that all existing instances for the specified resource type are removed before processing the specified file. The default is to not remove any resource. Multiple resources may be specified in a comma-separated list.

The valid resources are:

- ALL (clears all of the following resources)
- RULE
- MACHINE
- SCHEDULE
- ACTION
- GROUP
- DATE
- SPOOL
- EVENT
- PAGE
- IDLE

## Operand value lengths

Table 47 on page 82 lists the operand value lengths for the CONFIG command.

Table 47. CONFIG operand value lengths	
Operand	Length
CLEAR	1-120

Table 47. CONFIG operand value lengths (continued)	
Operand	Length
FILE	1-20
FM	1-2
FN	1-8
FT	1-8

## Example

An example is provided to help you understand how to use the CONFIG command to process the contents of a file containing Operations Manager commands, and to see the typical results of this command.

### Example 1: Use the CONFIG command to clear the contents of the current Operations Manager configuration before reloading any new entries from the specified configuration file.

An example of the CONFIG command is shown below:

```
GOMCMD OPMGRM1 CONFIG FILE 'OPMGRM1 CONFIG E' CLEAR ALL
```

This command clears all available resources that are currently loaded, and reloads the main configuration file.

## Usage notes

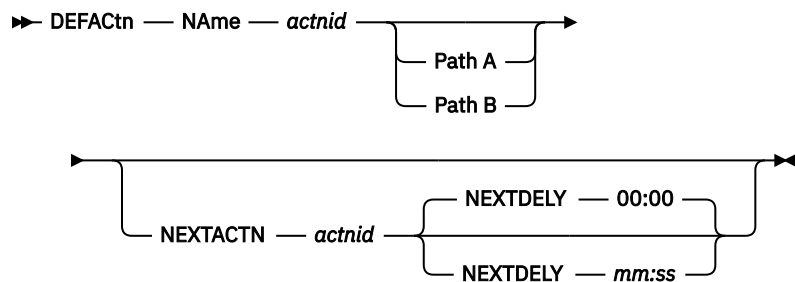
1. The CONFIG command can be used within a configuration file containing Operations Manager commands, allowing you to embed other configuration files. You can nest the CONFIG command within included files. There is no restriction on the depth of nested files. You can use this feature in the startup configuration file for the GOMMAIN command to organize your configuration data into multiple files.
2. If the FT operand is entered, the FN operand must be entered.
3. If the FM operand is entered, the FT operand must be entered.
4. The operands on one path are mutually exclusive with operands on other paths.
5. Use &SYSNAME to substitute the current system name for either the *fn* or *ft* value, or both values, in the FN, FT, and FILE operands. The &SYSNAME variable is replaced with the system name.
6. Use &SYSSIN to substitute the current SSI cluster for either the *fn* or *ft* value, or both values, in the FN, FT, and FILE operands. The &SYSSIN variable is replaced with the system SSI cluster name.

## Return codes

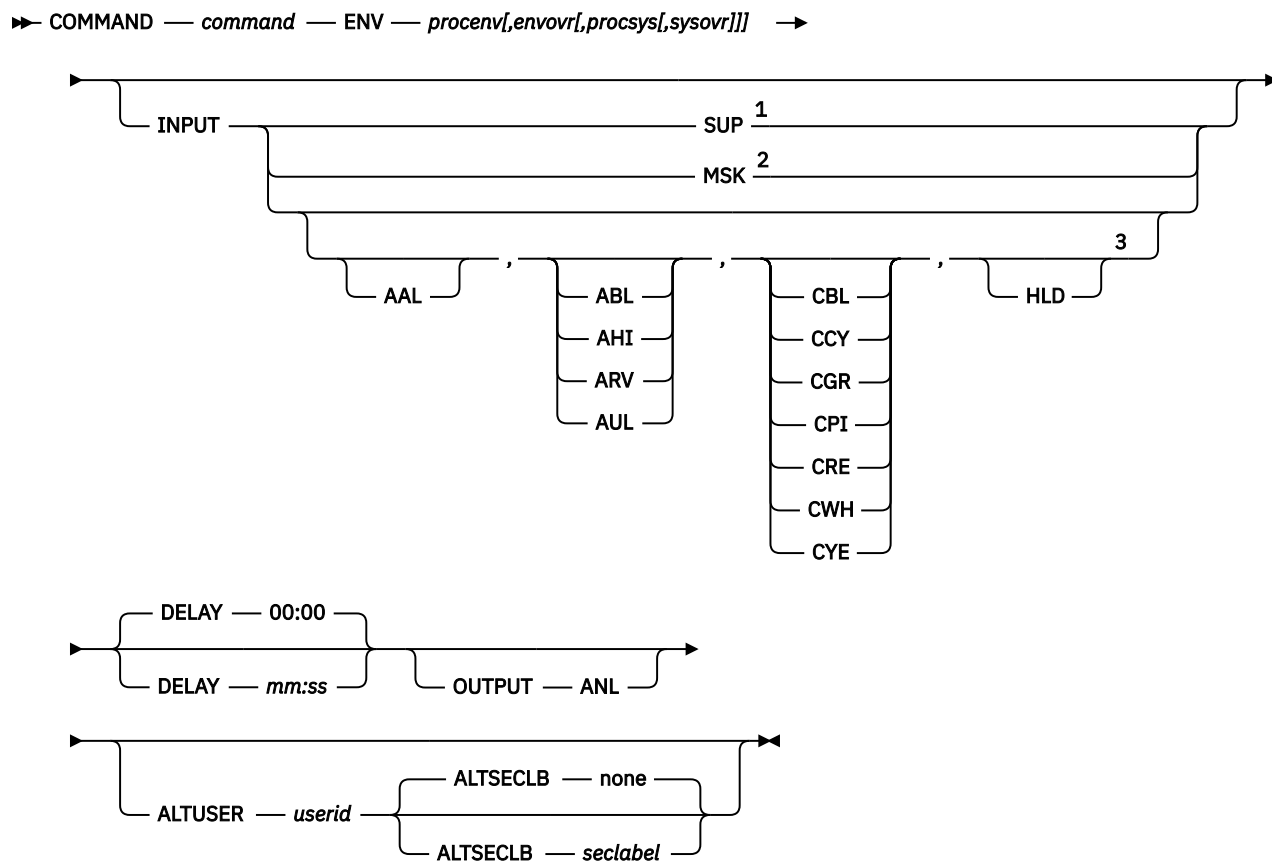
Table 48 on page 83 lists the return codes for the CONFIG command.

Table 48. CONFIG return codes	
Return code	Description
0	Finished correctly
-100	Error

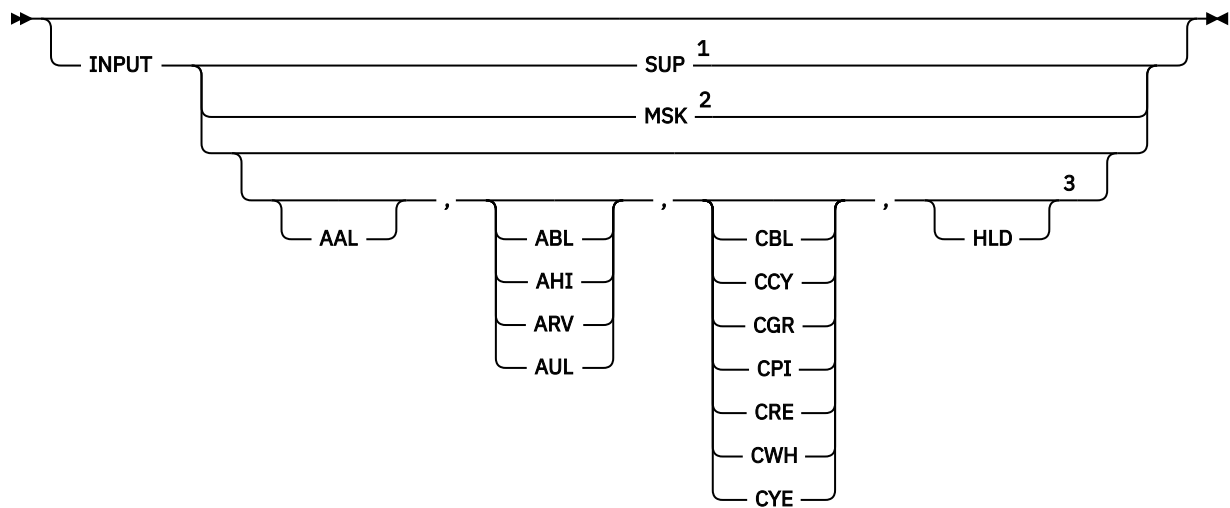
# DEFACTN



## Path A



## Path B



Notes:

- <sup>1</sup> If the SUP option is specified, no other options are allowed.
- <sup>2</sup> If the MSK option is specified, no other options are allowed.
- <sup>3</sup> Separate multiple options with a comma.

Figure 57. DEFACTN command syntax

## Authorization

The user issuing this command must be authorized for configuration commands.

## Purpose

Use the DEFACTN command to define an action to process as the result of a rule, schedule, or monitor.

## Operands

### NAME *actnid*

Specifies a name for the action.

An action with the name "NULL" is included in Operations Manager. The NULL action is equivalent to specifying DEFACTN NAME NULL and indicates that no action is to be performed. You cannot specify the value "NULL" for the NAME operand because it is already a built-in action.

### COMmand *command*

Identifies the command to be processed. This may be any CMS, CP, or Operations Manager command. The value may be entered as '*command*' if there is more than one blank-delimited word. Operations Manager builds the command allowing variable substitution. See [Appendix C, "Using substitution variables in commands,"](#) on page 183 for an explanation of the variables.

### ENV *procenv[,envovr[,procsys[,sysovr]]]*

Specifies the processing environment and system for this command. Also indicates if the environment and/or system can be overridden by a rule, schedule, or monitor which caused the command to run.

The valid values for *procenv* are:

#### GOM

Specifies that this is an Operations Manager command and is processed by the internal Operations Manager command processor.

#### LVM

Specifies that the command is processed by the local virtual machine (OPMGRM1). Use caution when specifying LVM. Any command or module that processes in the LVM preempts execution of

Operations Manager message processing and can possibly affect the performance and reliability of Operations Manager. It is recommended to use ENV (SVM) for most actions.

**SVM**

Specifies that the command is processed by any available action processing server.

**userid**

Specifies that the command is processed by only the action processing server indicated.

**wgrpid**

Specifies that the command is processed by only action processing servers participating in the work group.

The valid values for *envovr* are:

**N**

Specifies the command processing environment cannot be overridden by a rule, schedule, or monitor. This is the default.

**Y**

Specifies the command processing environment can be overridden by a rule, schedule, or monitor.

The only value for *procsys* is the system ID (*sysid*), which is described below:

**sysid**

Specifies that the command must be processed on the system named *sysid*. To specify the local system, use an asterisk (\*). The *sysid* should match the local system name, or the value of the NAME operand from a DEFIPCS command with APPL GOMRSCC.

The valid values for *sysovr* are:

**N**

Specifies the command processing system cannot be overridden by a rule, schedule, or monitor. This is the default.

**Y**

Specifies the command processing system can be overridden by a rule, schedule, or monitor.

**DELAY mm:ss**

Specifies an amount of time to delay before processing the command. Input is in minutes (mm) and seconds (ss).

**INPUT *inpwka***

Specifies what well-known action(s) Operations Manager should take upon the message that triggered the rule leading to this action.

**OUTPUT *outwka***

Specifies what well-known action(s) Operations Manager should take upon the message output of the command specified in the COMMAND operand. This only includes synchronous output. The only available action is ANL, which causes Operations Manager to process the synchronous output of the COMMAND operand through Operations Manager rules.

**NEXTACTN *actnid***

Specifies the name of an action to process upon completion of this action.

**NEXTDELY mm:ss**

Specifies an amount of time to delay the processing of NEXTACTN. The format is in minutes (mm) and seconds (ss).

**ALTUSER *userid***

Specifies the alternate user ID to be set for the service machine when processing the action. This operand is *not* valid for GOM or LVM environments. The alternate user ID is set using CP Diagnose '0xD4'. Refer to *z/VM CP Programming Services (SC24-6084)* for more information.

**ALTSECLB *seclabel***

Specifies the SECLABEL for the service machine when processing the action. This operand is valid *only* if ALTUSER is specified.

## Related reference

[DEFSRVM](#)

[ALTRCON](#)

[Application and settings interference](#)

## Example

Issue the DEFACTN command to define an action to process based on some additional criteria. The DEFACTN command can be used in combination with other commands, such as DEFSCHD to determine when the action should be performed.

### Example 1: Use the DEFACTN command to define an action to perform based on a specific schedule, as defined by the DEFSCHD command.

In this DEFACTN example, the action is sending a message (HELLO WORLD) to the monitored user (TS3163) at the scheduled time (WHEN NOW) listed in the DEFSCHD command.

```
GOMCMD OPMGRM1 DEFACTN NAME TEST1 COMMAND 'CP MSG TS3163 HELLO WORLD' ENV LVM
Ready; T=0.01/0.01 05:49:41
GOMCMD OPMGRM1 DEFSCHD NAME TES5SCHD ACTION TEST1 WHEN NOW
05:50:33 * MSG FROM OPMGRM1 : HELLO WORLD
Ready; T=0.01/0.01
05:50:33
```

Figure 58. Example of DEFACTN command with DEFSCHD "Now"

### Example 2: Use the DEFACTN command to define an action to perform based on a machine monitor rule, as defined in the DEFMMON command.

In this DEFACTN example, an action is defined to send a message when user MAINT640 is logged off.

```
GOMCMD OPMGRM1 DEFACTN NAME TEST5 COMMAND 'CP MSG MAINT640 IS LOGGED OFF' ENV LVM
Ready; T=0.01/0.01
02:16:45

GOMCMD OPMGRM1 DEFMMON NAME TEST9MON USER MAINT640 ACTION TEST5
Ready; T=0.01/0.01
02:17:29
```

Figure 59. Example of DEFACTN command with DEFMMON

A message is sent by Operations Manager indicating that this user, MAINT640, has logged off. An example is shown below.

```
02:19:00 * MSG FROM OPMGRM1 : MAINT640 IS LOGGED
OFF
```

### Example 3: Use the DEFACTN command to define an action to mask a line in the Operations Manager log upon matching the rule.

```
GOMCMD OPMGRM1 DEFACTN NAME ACT1 COMMAND 'CP MSG MAINT720 LINE MASKED!!' ENV LVM INPUT MSK
Ready; T=0.01/0.01 05:00:17
GOMCMD OPMGRM1 DEFRULE NAME RULE1 MATCH '*MACHINE*' ACTION ACT1 MASKNEXT 0
Ready; T=0.01/0.01 05:00:43
```

The Operations Manager log will contain the following.

```
08/22/22 05:02:20 GOMCMD0216L TS3159
"*****" VID=*MSG SRC=MASIUCV CLS=
08/22/22 05:02:20 GOMCMD0216L TS3159 "* -- Operations Manager Action ACT1 scheduled for
execution -- *" VID=*MSG SRC=MASC
08/22/22 05:02:20 GOMCMD0216L TS3159 "08/22/22 05:02:20 GOMCMD0903I 0 SCHEDULE(S) DEFINED"
VID=*MSG SRC=MASIUCV CLS=8
08/22/22 05:02:20 GOMCMD0216L TS3159 "08/22/22 05:02:20 GOMCMD0904I 2 ACTION(S) DEFINED"
VID=*MSG SRC=MASIUCV CLS=8
```

## Operand value lengths

Table 49 on page 88 lists the operand value lengths for the DEFACTN command.

Table 49. DEFACTN operand value lengths	
Operand	Length
ALTUSER	1-8
ALTSECLB	1-8
COMMAND	1-90
DELAY	5-5
INPUT	3-15
ENV	1-22
NAME	1-8
NEXTACTN	1-8
NEXTDELY	5
OUTPUT	3

## Usage notes

1. If OUTPUT is specified, COMMAND must be specified.
2. If COMMAND is specified, ENV must be specified.
3. If ENV is specified, COMMAND must be specified.
4. Substitution variables are available for all rules, schedules, and monitors in DEFACTN. For a list of valid substitution variables, refer to [Appendix C, “Using substitution variables in commands,”](#) on page 183.
5. The well-known input actions are:
  - AAL. Activates an audible alarm when the message is displayed.
  - ABL. Sets the extended display attribute to blink.
  - AHI. Sets the display attribute to high intensity.
  - ARV. Sets the extended display attribute to reverse video.
  - AUL. Sets the extended display attribute to underline.
  - CBL. Sets the extended display color to blue.
  - CCY. Sets the extended display color to cyan.
  - CGR. Sets the extended display color to green.
  - CPI. Sets the extended display color to pink.
  - CRE. Sets the extended display color to red.
  - CWH. Sets the extended display color to white.
  - CYE. Sets the extended display color to yellow.
  - HLD. Holds the message on the user's console until it is removed. The HLDATTR operand on DEFOPTN, the ALTRCON command, and the PF05 key in VIEWCON provide additional options for held lines.
  - SUP. Suppresses the message from the user's console.
    - a. If the SUP option is specified on the INPUT operand, no other options are allowed.
    - b. Only one extended display color is allowed.
    - c. Only one extended attribute is allowed.



- MSK. Masks the message in the Operations Manager log. MSK is only applicable for messages written to the Operations Manager log. If the MSK option is specified on the INPUT operand, no other options are allowed. Specify additional DEFACTN statements to take additional actions.
6. The only available output action is ANL, which causes Operations Manager to process the synchronous output of the COMMAND operand through Operations Manager rules. If an action is defined with OUTPUT(ANL), the output from the command is captured in the console data space named GOMACTN.
  7. If ALTUSER is specified, the user ID OPMGRM1 or any user ID running GOMMAIN, must have PRIV B to operate correctly. For more information on privilege classes, refer to [Chapter 2, “Configuring Operations Manager,”](#) on page 5.
  8. An action with the name "NULL" is included in Operations Manager. The NULL action is equivalent to specifying "DEFACTN NAME NULL" and indicates that no action is to be performed. You cannot specify the value "NULL" for the NAME operand because it is already a built-in action.
  9. If the COMMAND operand is prefixed with the value 'CP ' (upper case CP, followed by a space), the command is processed through the CP diagnose x'0008' interface. The diagnose x'0008' interface may be used to send case-sensitive data to z/VM guests, such as Linux. The DEFOPTN value for the MIXCASE operand must also be 'Y' to use case-sensitive data.

#### Related reference

[ALTRCON](#)

[DEFOPTN](#)

[VIEWCON](#)

[Using substitution variables in commands](#)

## Return codes

Table 50 on page 89 lists the return codes for the DEFACTN command.

Table 50. DEFACTN return codes	
Return code	Description
0	Finished correctly
-100	Error

## DEFDATE

➤ DEFDATE — NAmE — *dateid* — HOLIDAY — 'yyyy/mm/dd-hh:mm yyyy/mm/dd-hh:mm' ➤

*Figure 60. DEFDATE command syntax*

## Authorization

The user issuing this command must be authorized for configuration commands.

## Purpose

Use the DEFDATE command to define a date as a holiday. These dates may then be included or excluded from rules, monitors, and schedules.

## Operands

**NAmE** *dateid*

Specifies the name of the holiday.

**HOLIDAY** 'yyyy/mm/dd-hh:mm yyyy/mm/dd-hh:mm'

Defines the beginning date (year, month, day) and time (hours, minutes) of the holiday, and the ending date/time.

**Related reference**

[DEFEMON](#)

[DEFMMON](#)

[DEFSCHD](#)

[DEFSMON](#)

## Operand value lengths

Table 51 on page 90 lists the operand value lengths for the DEFDATE command.

Table 51. DEFDATE operand value lengths	
Operand	Length
HOLIDAY	33
NAME	1-8

## Example

An example is provided to help you understand how to use the DEFDATE command, and shows the typical results of this command.

**Example 1: Use the DEFDATE command to define a date as a HOLIDAY. Specify the beginning and ending dates in the format 'yyyy/mm/dd-hh:mm yyyy/mm/dd-hh:mm'.**

```
GOMCMD OPMGRM1 DEFDATE NAME XMAS HOLIDAY '2020/12/25-00:00 2020/12/25-23:55'  
Ready; T=0.01/0.01 06:25:41
```

Figure 61. DEFDATE command issued

## Usage notes

None

## Return codes

Table 52 on page 90 lists the return codes for the DEFDATE command.

Table 52. DEFDATE return codes	
Return code	Description
0	Finished correctly
-100	Error

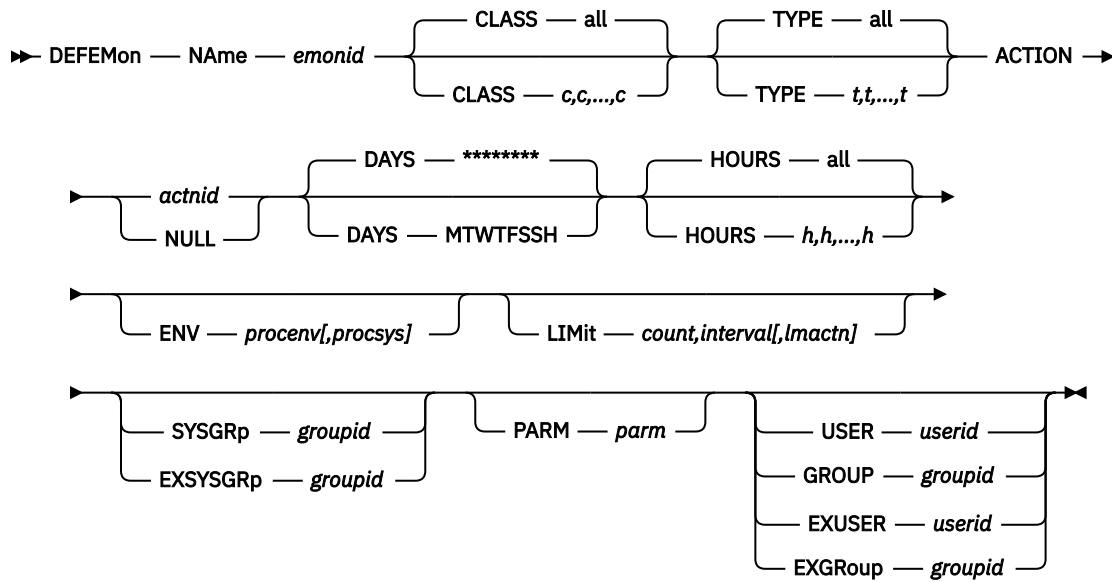


Figure 62. DEFEMON command syntax

## Authorization

The user issuing this command must be authorized for configuration commands.

## Purpose

Use the DEFEMON command to define an event monitor. Events from the CP system service, \*VMEVENT, are captured and compared against defined event monitors. If conditions for the event monitor are met, the associated action is scheduled for processing.

## Operands

### NAme *emonid*

Specifies the name of the event monitor.

### CLASS *c,c,...,c*

Specifies the classes of events to monitor. The values for CLASS are listed in the description of the \*VMEVENT system service in the *CP Programming Services* (SC24-6179) documentation. The default value is all classes.

### TYPE *t,t,...,t*

Specifies the types of events to monitor. The values for TYPE are listed in the description of the \*VMEVENT system service in the *CP Programming Services* (SC24-6179) documentation. The default value is all types.

### ACTION *actnid*

Specifies the name of the associated action. This action must have been previously defined using the DEFACTN command or the built-in action NULL.

An action with the name "NULL" is included in Operations Manager. The NULL action is equivalent to specifying DEFACTN NAME NULL and indicates that no action is to be performed. This is useful for rules and monitors that require no action initially, but may require action due to a LIMIT being reached or an idle monitor (via DEFIMON) being triggered.

### DAYS *MTWTFSSH*

Specifies the days of the week (Monday-Sunday) and if the event is active on holidays. For the event to be active on a day of the week and holiday, the corresponding location must contain the asterisk

(\*) symbol. Issue the DEFDATE command to define holidays. The default is all days and holidays (\*\*\*\*\*).

**HOURS *h,h,...,h***

Specifies the hours of the day the event is active. Any values from 0 to 23 may appear in the comma-separated list. Up to 24 values may be present. The default is all hours.

**ENV *procenv[,procsys]***

Specifies that the processing environment and system of the action is to be overridden with the values assigned to *procenv* and *procsys*. See “DEFACTN” on page 84 for more information on overriding the processing environment and system, and valid values for *procenv* and *procsys*.

**LIMIT *count,interval[,lmactn]***

Specifies that the processing of the monitor's ACTION is limited to *count* times within *interval* period of time. Optionally, this command processes the alternate action specified by *lmactn* if the limit condition is encountered.

- The value for *count* must be from 0 to 999999.
- When specified as a single positive integer, the value for *interval* must be from 0 to 999999 seconds. You can also use the format of ddd-hh:mm:ss, which specifies the interval value in days (ddd), hours (hh), minutes (mm), and seconds (ss). The maximum value for this input format is 999-23:59:59.

For example, a LIMIT of 4 times in 10 seconds could be specified as:

```
LIMIT '4,10'    OR    LIMIT '4,000-00:00:10'
```

- A value of 0 for *count* and *interval* indicates that no processing limit should be applied. If count or interval are 0, both count and interval must be 0, and the *lmactn* value omitted. If count and interval are non-zero, the alternate action *lmactn* may be specified. The action *lmactn* is only processed once if the limit condition is encountered within an interval.

**Note:** If the LIMIT operand is omitted, the default count and interval are obtained from the DEFOPTN command, EMONLIM operand.

**SYSGRP *groupid***

Specifies that this monitor is only applicable for systems where the local system name is in the group named *groupid*.

**EXSYSGRP *groupid***

Specifies that this monitor is only applicable for systems where the local system name is *not* in the group named *groupid*.

**PARM *parm***

Specifies the parameter or list of parameters that are passed to the ACTION when it is scheduled to run. The value(s) may be entered as *parm*, but must be enclosed in quotation marks ('parm') if there is more than one blank-delimited value. The DEFACTN command can use the parameter through the &p substitution variable. The parameter is also available to any chained action routine that is defined in the NEXTACTN operand on the DEFACTN command.

**USER *userid***

Specifies this event is only applicable for events generated for *userid*.

**GROUP *groupid***

Specifies this event is only applicable for events generated for users in the group named *groupid*.

**EXUSER *userid***

Specifies this event is only applicable for events NOT generated for *userid*. EXUSER may also be used to specify an identifier for events that do not involve a *userid*.

**EXGROUP *groupid***

Specifies this event is only applicable for events NOT generated for users in the group named *groupid*.

**Related reference**

[DEFACTN](#)

[DEFDATE](#)

## Operand value lengths

Table 53 on page 93 lists the operand value lengths for the DEFEMON command.

Table 53. DEFEMON operand value lengths	
Operand	Length
ACTION	1-8
CLASS	1-120
DAYS	8
ENV	1-8
EXGROUP	1-8
EXUSER	1-8
EXSYSGRP	1-8
GROUP	1-8
HOURS	1-72
LIMIT	3-28
NAME	1-8
PARM	1-120
SYSGRP	1-8
TYPE	1-120
USER	1-8

## Examples

An example is provided to help you understand how to use the DEFEMON command, and shows the typical results of this command. Events from the CP system service, \*VMEVENT, are captured and compared against defined event monitors. If conditions for the event monitor are met, the associated action is scheduled for processing.

### Example 1: Use the DEFEMON command to define an event monitor named EMONITOR.

In the example below, an event monitor is defined for user TS3163 with a TYPE 1 event, which is a "LOGOFF" event. When user ID TS3163 logs off, an action named SAMPLE is triggered, which will issue a message "USER ID TS3163 IS LOGGED OFF" on the MAINT640 screen.

```
GOMCMD OPMGRM1 DEFACTN NAME SAMPLE COMMAND 'CP MSG MAINT640 USER ID TS3163 IS LOGGED OFF' ENV
LVM
Ready; T=0.01/0.01 02:17:24

GOMCMD OPMGRM1 DEFEMON NAME EMONITOR TYPE 1 USER TS3163 ACTION SAMPLE
Ready; T=0.01/0.01 02:17:53

MAINT640 AT ZVM6KT02 VIA *          09/16/19 02:18:57 CDT          MONDAY
Ready; T=0.01/0.01 02:18:57
02:19:03 * MSG FROM OPMGRM1 : USER ID TS3163 IS LOGGED OFF
```

Figure 63. Example of DEFEMON command with DEFACTN command

### Example 2: Use the DEFEMON command to define an event monitor, EVENTMON, using TYPE 0.

In the example below, an event monitor is defined for user MAINT640 with a TYPE 0 event, which is a "LOGON" event. When user MAINT640 logs on to the system, an action named SAMPLE is triggered, which will issue a message "USER ID MAINT640 HAS LOGGED ON SUCCESSFULLY" to the user TS3163.

```
GOMCMD OPMGRM1 DEFACN NAME SAMPLE COMMAND 'CP MSG TS3163 USER ID MAINT640 HAS LOGGED ON
SUCCESSFULLY' ENV LVM
Ready; T=0.01/0.01 03:53:43

GOMCMD OPMGRM1 DEFEMON NAME EVENTMON TYPE 0 USER MAINT640 ACTION SAMPLE
Ready; T=0.01/0.01 03:54:49

03:55:02 * MSG FROM OPMGRM1 : USER ID MAINT640 HAS LOGGED ON SUCCESSFULLY
```

## Usage notes

1. The USER, GROUP, EXUSER, and EXGROUP operands are mutually exclusive.
2. The SYSGRP operand and the EXSYSGRP operand are mutually exclusive.
3. The message field passed to the action may contain a variety of parameter details. For more information on action substitution variables, see [Appendix C, "Using substitution variables in commands,"](#) on page 183.
4. The message field passed to the action contains the following fields:
  - &0 – Constant "VMEVENT"
  - &1 – Name of the event monitor
  - &2 – Name of the event monitor's action
  - &3 – User ID or identifier which originated this event
  - &4 – Event type
  - &5 – Event class
  - &6-&n – Any additional information passed for a particular event class/type.
5. The message field that is passed to the *lmtactn* action of the LIMIT operand contains the following fields:
  - &0 – Identifies the constant LIMIT.
  - &1 – Identifies the constant EVENT.
  - &2 – Indicates the name of the event monitor.
  - &3 – Indicates the LIMIT count.
  - &4 – Indicates the LIMIT interval.
  - &5 – Identifies the total number of times that the normal action was suppressed.
  - &6-&n – Specifies the message field passed to the previous processing ACTION.

For more information on these substitution variables, see [Appendix C, "Using substitution variables in commands,"](#) on page 183.

## Return codes

Table 54 on page 94 lists the return codes for the DEFEMON command.

Table 54. DEFEMON return codes	
Return code	Description
0	Finished correctly
-100	Error

# DEFIMON

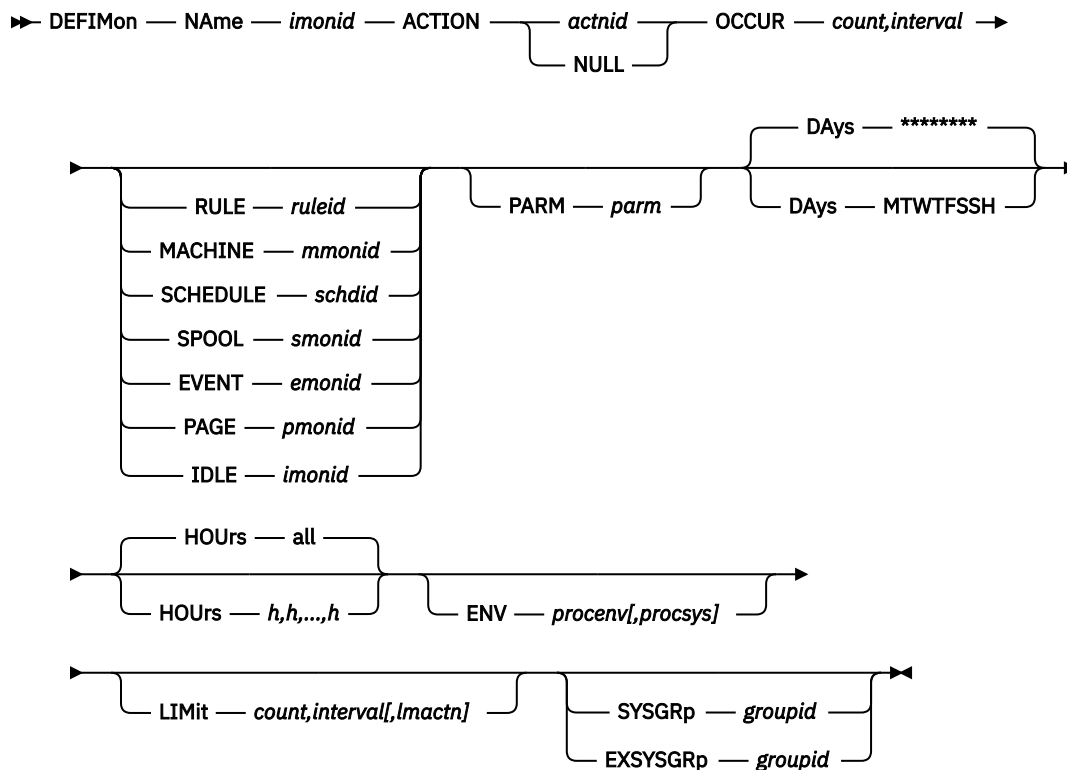


Figure 64. DEFIMON command syntax

## Authorization

The user issuing this command must be authorized for configuration commands.

## Purpose

Use the DEFIMON command to define an idle monitor. A check is performed to determine if the RULE, MACHINE, SCHEDULE, SPOOL, EVENT, PAGE, or IDLE resource is triggered a specified number of times within a specified time frame. If a resource is **not triggered**, the action associated with the idle monitor is scheduled to run. See the [“Usage notes” on page 98](#) for details of the message passed to the action.

## Operands

### NAME *imonid*

Specifies the name of the idle monitor.

### ACTION *actnid*

Specifies the name of the action to process when a resource is not triggered. This action must have been previously defined using the DEFACTN command or the built-in action NULL.

An action with the name "NULL" is included in Operations Manager. The NULL action is equivalent to specifying DEFACTN NAME NULL and indicates that no action is to be performed. This is useful for rules and monitors that require no action initially, but may require action due to a LIMIT being reached or an idle monitor (via DEFIMON) being triggered.

### OCCUR *count, interval*

Specifies that the processing of the monitored resource should happen *count* times, within *interval* minutes. The values for *count* and *interval* must be from 1 to 999999.

### RULE *ruleid*

Specifies to monitor the rule *ruleid*.

**MACHINE *mmonid***

Specifies to monitor the machine monitor *mmonid*.

**SCHEDULE *schdid***

Specifies to monitor the schedule *schdid*.

**SPOOL *smonid***

Specifies to monitor the spool monitor *smonid*.

**EVENT *emonid***

Specifies to monitor the event monitor *emonid*.

**PAGE *pmonid***

Specifies to monitor the page space monitor *pmonid*.

**IDLE *imonid***

Specifies to monitor the idle monitor *imonid*.

**PARM *parm***

Specifies the parameter or list of parameters that are passed to the ACTION when it is scheduled to run. The value(s) may be entered as *parm*, but must be enclosed in quotation marks ('*parm*') if there is more than one blank-delimited value. The DEFACTN command can use the parameter through the &p substitution variable. The parameter is also available to any chained action routine that is defined in the NEXTACTN operand on the DEFACTN command.

**DAYS *MTWTFSSH***

Specifies the days of the week (Monday-Sunday) and if the monitor is active on holidays. For the monitor to be active on a day of the week and holiday, the corresponding location must contain the asterisk (\*) symbol. The default is all days and holidays (\*\*\*\*\*). See [“DEFDATE” on page 89](#) to define holidays.

**HOURS *h,h,...,h***

Specifies the hours of the day the monitor is checked. Any value from 0 to 23 can be used in a comma-separated list. Up to 24 values can be present. The default is all hours.

**ENV *procenv [,procsys]***

Specifies that the processing environment and the processing system of the ACTION is to be overridden with the supplied values for *procenv* and *procsys*. See [“DEFACTN” on page 84](#) for additional information on overriding the processing environment and processing system.

**LIMIT *count,interval[,lmactn]***

Specifies that the processing of the monitor's ACTION is limited to *count* times within *interval* period of time. Optionally, this command processes the alternate action specified by *lmactn* if the limit condition is encountered.

- The value for *count* must be from 0 to 999999.
- When specified as a single positive integer, the value for *interval* must be from 0 to 999999 seconds. You can also use the format of *ddd-hh:mm:ss*, which specifies the interval value in days (*ddd*), hours (*hh*), minutes (*mm*), and seconds (*ss*). The maximum value for this input format is 999-23:59:59.

For example, a LIMIT of 4 times in 10 seconds could be specified as:

```
LIMIT '4,10'    OR    LIMIT '4,000-00:00:10'
```

- A value of 0 for *count* and *interval* indicates that no processing limit should be applied. If *count* or *interval* are 0, both *count* and *interval* must be 0, and the *lmactn* value omitted. If *count* and *interval* are non-zero, the alternate action *lmactn* may be specified. The action *lmactn* is only processed once if the limit condition is encountered within an interval.

**Note:** If the LIMIT operand is omitted, the default *count* and *interval* are obtained from the DEFOPTN command, EMONLIM operand.

**SYSGRP *groupid***

Specifies that this monitor is only applicable for systems where the local system name is in the group named *groupid*.



**EXSYSGRP *groupid***

Specifies that this monitor is only applicable for systems where the local system name is *not* in the group named *groupid*.

**Related reference**

[DEFGROUP](#)

## Operand value lengths

Table 55 on page 97 lists the operand value lengths for the DEFIMON command.

Table 55. DEFIMON operand value lengths	
Operand	Length
ACTION	1-8
DAYS	8
ENV	1-8
EVENT	1-8
EXSYSGRP	1-8
HOURS	1-72
IDLE	1-8
LIMIT	3-28
MACHINE	1-8
NAME	1-8
OCCUR(m,n)	1-6, 1-6
PAGE	1-8
PARM	1-120
RULE	1-8
SCHEDULE	1-8
SPOOL	1-8
SYSGRP	1-8

## Example

An example is provided to help you understand how to use the DEFIMON command, and shows the typical results of this command.

**Example 1: Use the DEFIMON command to define an idle monitor, IDMON.**

In the example below, the following tasks are performed:

1. Schedule TESTSCHD is defined before the IDLE monitor is defined.
2. Action IACT is defined before the definition of IDLE monitor IDMON.
3. IDLE monitor, IDMON, is defined, which checks if the schedule TESTSCHD is triggered 10 times in 1 minute. If not, action "IACT" is run.

```

GOMCMD OPMGRM1 DEFSCHD NAME TESTSCHD ACTION ACT EVERY 00:01
Ready; T=0.01/0.01 06:13:53

GOMCMD OPMGRM1 DEFACTN NAME IACT COMMAND 'CP MSG TS3163 IDLE MONITOR' ENV LVM
Ready; T=0.01/0.01 06:18:05

GOMCMD OPMGRM1 DEFIMON NAME IDMON ACTION IACT OCCUR 10,1 SCHEDULE TESTSCHD
Ready; T=0.01/0.01 06:29:47

RESULT: -
06:31:30 * MSG FROM OPMGRM1 : IDLE MONITOR
06:32:30 * MSG FROM OPMGRM1 : IDLE MONITOR
06:33:30 * MSG FROM OPMGRM1 : IDLE MONITOR
06:34:30 * MSG FROM OPMGRM1 : IDLE MONITOR
06:35:30 * MSG FROM OPMGRM1 : IDLE MONITOR
06:36:30 * MSG FROM OPMGRM1 : IDLE MONITOR
06:37:30 * MSG FROM OPMGRM1 : IDLE MONITOR
06:38:30 * MSG FROM OPMGRM1 : IDLE MONITOR
06:39:30 * MSG FROM OPMGRM1 : IDLE MONITOR

```

Figure 65. Example of DEFIMON command with DEFACTN command

## Usage notes

1. Only one of the operands RULE, MACHINE, SCHEDULE, SPOOL, EVENT, PAGE, or IDLE must be specified.
2. The message field that is passed to the action contains the following fields:
  - &0 — Constant "IDLE"
  - &1 — Name of the idle monitor.
  - &2 — Name of the idle monitor's action.
  - &3 — Type of resource that is monitored: RULE, MACHINE, SCHEDULE, SPOOL, EVENT, PAGE, or IDLE
  - &4 — Name of the monitored resource.
  - &5 — Number of times the monitored resource was triggered within the interval specified on the OCCUR operand.

For more information on action substitution variables, see [Appendix C, "Using substitution variables in commands,"](#) on page 183.

3. The message field that is passed to the *lmtactn* action of the LIMIT operand contains the following fields:
  - &0 — Identifies the constant LIMIT.
  - &1 — Identifies the constant IDLE.
  - &2 — Indicates the name of the idle monitor.
  - &3 — Indicates the LIMIT count.
  - &4 — Indicates the LIMIT interval.
  - &5 — Identifies the total number of times that the normal action was suppressed.
  - &6-&n — Specifies the message field passed to the previous processing ACTION.

For more information on these substitution variables, see [Appendix C, "Using substitution variables in commands,"](#) on page 183.

4. The SYSGRP operand and the EXSYSGRP operand are mutually exclusive.

## Return codes

[Table 56 on page 99](#) lists the return codes for the DEFIMON command.

Table 56. DEFIMON return codes	
Return code	Description
0	Finished correctly
-100	Error

## DEFGROUP

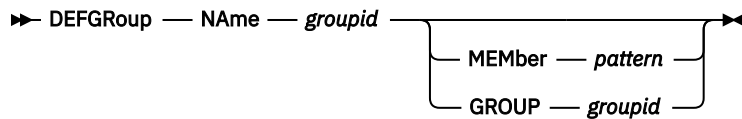


Figure 66. DEFGROUP command syntax

### Authorization

The user issuing this command must be authorized for configuration commands.

### Purpose

Use the DEFGROUP command to define a group of members to which rules, schedules, and monitors can apply. This command adds members to a group.

After a group is defined, several commands and operands can refer to the group by its group name. This includes the GROUP, EXGROUP, SYSGRP, and EXSYSGRP operands, some of which are available on the DEFEMON, DEFIMON, DEFMMON, DEFPMON, DEFRULE, DEFSCHD, and DEFSMON commands. Depending on how the group is used on these commands, it can refer to groups of users or groups of system names.

### Operands

#### **NAME *groupid***

Specifies a name for the group

#### **MEMber *pattern***

Specifies a member to add to your group. Using the MEMBER operand allows groups to be used as a generalized collection of members and searched for inclusion/exclusion as needed. The *pattern* may contain wildcard matching information.

#### **GROUP *groupid***

Specifies a group of members to add to the group.

#### **Related concepts**

[Wildcard pattern matching](#)

Wildcard pattern matching is used by several of the commands. Wildcards allow you to search for and process characters that are similar.

#### **Related reference**

[AUTH](#)

[DEFDSPS](#)

[DEFEMON](#)

[DEFIMON](#)

[DEFMMON](#)

[DEFPMON](#)

[DEFRULE](#)

[DEFSCHD](#)

[DEFSMON](#)

[DEFVIEW](#)  
[DELDSPS](#)  
[DELVIEW](#)  
[REVOKE](#)

## Operand value lengths

Table 57 on page 100 lists the operand value lengths for the DEFGROUP command.

Table 57. DEFGROUP operand value lengths	
Operand	Length
MEMBER	1-8
NAME	1-8
USER	1-8

## Example

An example is provided to help you understand how to use the DEFGROUP command, and shows the typical results of this command.

**Example 1: Use the DEFGROUP command to define a group of members (RANGERS) to which rules, schedules, and monitors will apply.**

The example below shows a group named **RANGERS** and each member (**MAINT640**, **TS3163** and **TS3163A**) that is being added to the group.

```
GOMCMD OPMGRM1 DEFGROUP NAME RANGERS MEMBER MAINT640
Ready; T=0.01/0.01 06:43:49
GOMCMD OPMGRM1 DEFGROUP NAME RANGERS MEMBER TS3163
Ready; T=0.01/0.01 06:44:57
GOMCMD OPMGRM1 DEFGROUP NAME RANGERS MEMBER TS3163A
Ready; T=0.01/0.01 06:48:50
```

Figure 67. DEFGROUP command issued

## Usage notes

1. Either the MEMBER operand or the GROUP operand must be specified.
2. The MEMBER operand and GROUP operand are mutually exclusive.
3. Use a wildcard pattern to add multiple, similarly named members to a group . For example, if you have several members whose IDs begin with "ZVMSSI" and you want to add all of those members to the "SSILPARS" group, enter the following command:

```
DEFGROUP NAME SSILPARS MEMBER ZVMSSI*
```

For more information on wildcard pattern matching, see [Appendix B, "Wildcard pattern matching,"](#) on page 181.

4. If you want to add multiple member IDs to a group for which "wildcarding" is not possible, you can specify one DEFGROUP command for each member ID. For example, to add OPERATOR and TCPIP to group GROUP1, specify two DEFGROUP commands.

```
DEFGROUP NAME GROUP1 MEMBER OPERATOR
DEFGROUP NAME GROUP1 MEMBER TCPIP
```

## Return codes

Table 58 on page 101 lists the return codes for the DEFGROUP command.

Table 58. DEFGROUP return codes	
Return code	Description
0	Finished correctly
-100	Error

## DEFMMON

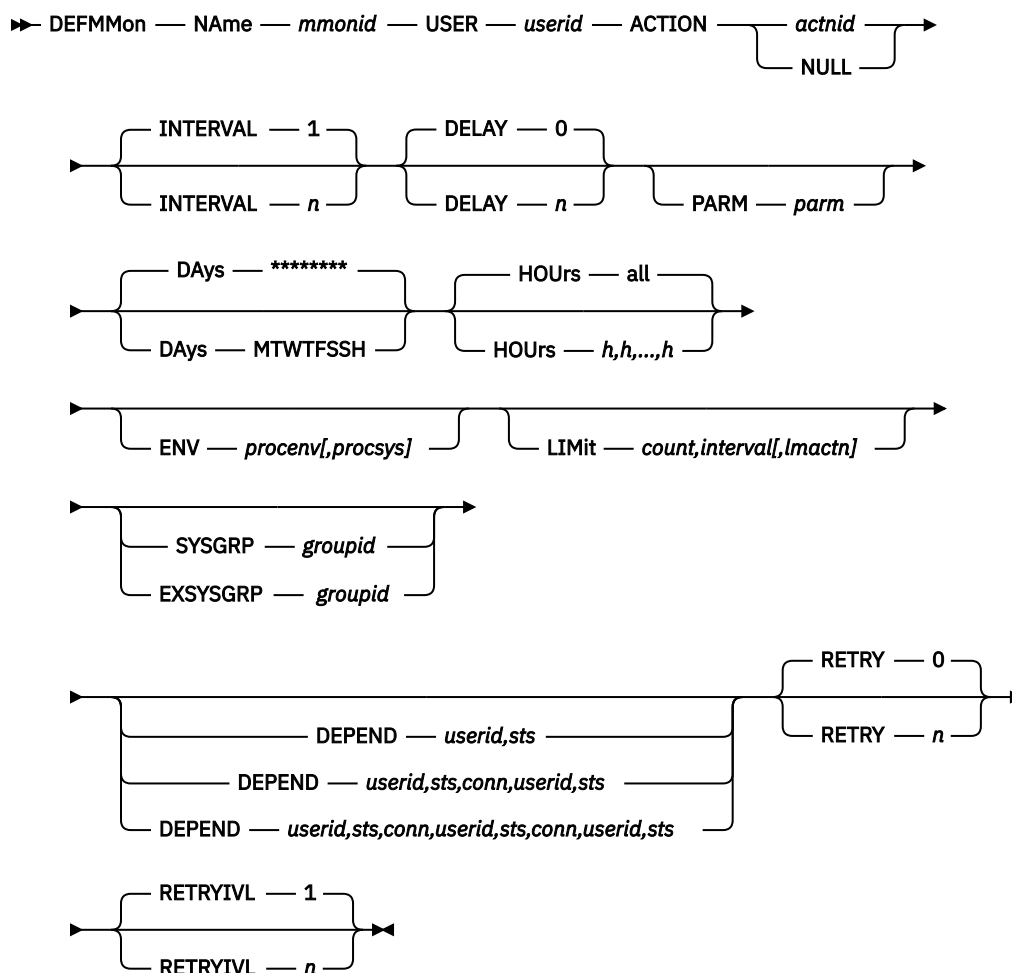


Figure 68. DEFMMON command syntax

## Authorization

The user issuing this command must be authorized for configuration commands.

## Purpose

Use the DEFMMON command to define a virtual machine to monitor. A check is performed to determine if the virtual machine is logged off. If it is, the specified action is scheduled to be processed, which may include a command required to bring the virtual machine to a logged on status.

## Operands

### NAME *mmonid*

Specifies the name of the machine monitor.

**USER *userid***

Specifies the user ID of the virtual machine to monitor.

**ACTION *actnid***

Specifies the name of the action to process to bring the virtual machine to a logged on status. This action must have been previously defined using the DEFACTN command or the built-in action NULL.

An action with the name "NULL" is included in Operations Manager. The NULL action is equivalent to specifying DEFACTN NAME NULL and indicates that no action is to be performed. This is useful for rules and monitors that require no action initially, but may require action due to a LIMIT being reached or an idle monitor (via DEFIMON) being triggered.

**INTERVAL *n***

Specifies how often, in minutes, to check the virtual machine status. The default is 1 minute.

**DELAY *n***

Specifies how long to wait, in minutes, after the monitor is defined to check the virtual machine status. The default is 0 (check immediately).

**PARM *parm***

Specifies the parameter or list of parameters that is passed to the ACTION when it is scheduled to run. The value(s) may be entered as *parm*, but must be enclosed in quotation marks (' *parm* ') if there is more than one blank-delimited value. The DEFACTN command can use the parameter through the & substitution variable. The parameter is also available to any chained action routine that is defined in the NEXTACTN operand on DEFACTN.

**DAYS *MTWTFSSH***

Specifies the days of the week (Monday-Sunday) and if the monitor is checked on holidays. For the monitor to be active on a day of the week and holiday, the corresponding location must contain the asterisk (\*) symbol. The default is all days and holidays (\*\*\*\*\*). For information on how to define holidays, see [“DEFDATE” on page 89](#).

**HOURS *h,h,...,h***

Specifies the hours of the day the monitor is checked. Any value 0 to 23 may appear in the comma-separated list. Up to 24 values may be present. The default is all hours.

**ENV *procenv[,procsys]***

Specifies that the processing environment and system of the action is to be overridden with the values assigned to *procenv* and *procsys*. See [“DEFACTN” on page 84](#) for more information on overriding the processing environment and system, and valid values for *procenv* and *procsys*.

**LIMIT *count,interval[,lmactn]***

Specifies that the processing of the monitor's ACTION is limited to *count* times within *interval* period of time. Optionally, this command processes the alternate action specified by *lmactn* if the limit condition is encountered.

- The value for *count* must be from 0 to 999999.
- When specified as a single positive integer, the value for *interval* must be from 0 to 999999 seconds. You can also use the format of *ddd-hh:mm:ss*, which specifies the interval value in days (*ddd*), hours (*hh*), minutes (*mm*), and seconds (*ss*). The maximum value for this input format is 999-23:59:59.

For example, a LIMIT of 4 times in 10 seconds could be specified as:

```
LIMIT '4,10'    OR    LIMIT '4,000-00:00:10'
```

- A value of 0 for *count* and *interval* indicates that no processing limit should be applied. If count or interval are 0, both count and interval must be 0, and the *lmactn* value omitted. If count and interval are non-zero, the alternate action *lmactn* may be specified. The action *lmactn* is only processed once if the limit condition is encountered within an interval.

**Note:** If the LIMIT operand is omitted, the default count and interval are obtained from the DEFOPTN command, EMONLIM operand.

**SYSGRP *groupid***

Specifies that this monitor is only applicable for systems where the local system name is in the group named *groupid*.

**EXSYSGRP *groupid***

Specifies that this monitor is only applicable for systems where the local system name is *not* in the group named *groupid*.

**DEPEND *userid, sts*****DEPEND *userid, sts, conn, userid, sts*****DEPEND *userid, sts, conn, userid, sts, conn, userid, sts***

Specifies up to three virtual machines and their status before the monitor is checked.

- *userid* — Specifies the user ID of the virtual machine to check.
- *sts* — Specifies the desired status of the virtual machine:
  - A — Active. The virtual machine is logged on (or running disconnected).
  - I — Inactive. The virtual machine is logged off.
- *conn* — Specifies the logical connector:
  - | (vertical bar) — or/exclude
  - & (ampersand) — and/include

The logical connectors are evaluated left to right and have no hierarchy. The default is no dependencies.

**RETRY *n***

Specifies the number of times to retry the action if the monitored virtual machine is not brought to a logged on status. Each INTERVAL that does not bring the virtual machine to a logged on status is considered a retry. When the number of retries is exceeded, the monitor checking interval is advanced by the number of minutes in the RETRYIVL operand. This is useful if a virtual machine does not attain or remain in the logged on status. It can reduce the resource consumption used by a continuously failing virtual machine. The default is 0.

**RETRYIVL *n***

Specifies the number of minutes to elapse before the monitor is checked if the retry is exceeded. If specified, it must be greater than or equal to INTERVAL. The default is the value of INTERVAL.

**Related reference**

[DEFACTN](#)

[DEFDATE](#)

[DEFGROUP](#)

## Operand value lengths

Table 59 on page 103 lists the operand value lengths for the DEFMMON command.

Table 59. DEFMMON operand value lengths	
Operand	Length
ACTION	1-8
DAYS	8
DELAY	1-6
DEPEND	3-36
ENV	1-8
EXSYSGRP	1-8
HOURS	1-72

Table 59. DEFMMON operand value lengths (continued)	
Operand	Length
INTERVAL	1-6
LIMIT	3-28
NAME	1-8
PARM	1-120
RETRY	1-6
RETRYIVL	1-6
SYSGRP	1-8
USER	1-8

## Example

An example is provided to help you understand how to use the DEFMMON command, and shows the typical results of this command.

### Example 1: Use the DEFMMON command to monitor the logon status of user ID MAINT640.

A check is performed every minute (the default) to determine if the virtual machine is logged off. If it is, the TEST5 action that is defined with the DEFACTN command is scheduled to process and displays a message, "MAINT640 IS LOGGED OFF".

```
GOMCMD OPMGRM1 DEFACTN NAME TEST5 COMMAND 'CP MSG MAINT640 IS LOGGED OFF' ENV LVM
Ready; T=0.01/0.01
02:16:45

GOMCMD OPMGRM1 DEFMMON NAME TEST9MON USER MAINT640 ACTION TEST5
Ready; T=0.01/0.01 02:17:29
```

Figure 69. DEFACTN and DEFMMON commands

When user MAINT640 logs off, a message displays to inform you of that action.

```
02:19:00 * MSG FROM OPMGRM1 : MAINT640 IS LOGGED OFF
```

## Usage notes

1. The message field passed to the action contains the following fields:

- &0 — Constant "MACHINE"
- &1 — The name of the machine monitor
- &2 — The name of the machine monitor's action
- &3 — The user ID that caused the action

For more information on action substitution variables, see [Appendix C, "Using substitution variables in commands,"](#) on page 183.

2. The SYSGRP operand and the EXSYSGRP operand are mutually exclusive.

3. The message field that is passed to the *lmtactn* action of the LIMIT operand contains the following fields:

- &0 — Identifies the constant LIMIT.
- &1 — Identifies the constant MACHINE.



- &2 — Indicates the name of the machine monitor.
- &3 — Indicates the LIMIT count.
- &4 — Indicates the LIMIT interval.
- &5 — Identifies the total number of times that the normal action was suppressed.
- &6-&n — Specifies the message field passed to the previous processing ACTION.

For more information on these substitution variables, see [Appendix C, “Using substitution variables in commands,”](#) on page 183.

## Return codes

Table 60 on page 105 lists the return codes for the DEFMMON command.

Table 60. DEFMMON return codes	
Return code	Description
0	Finished correctly
-100	Error

## DEFPMON

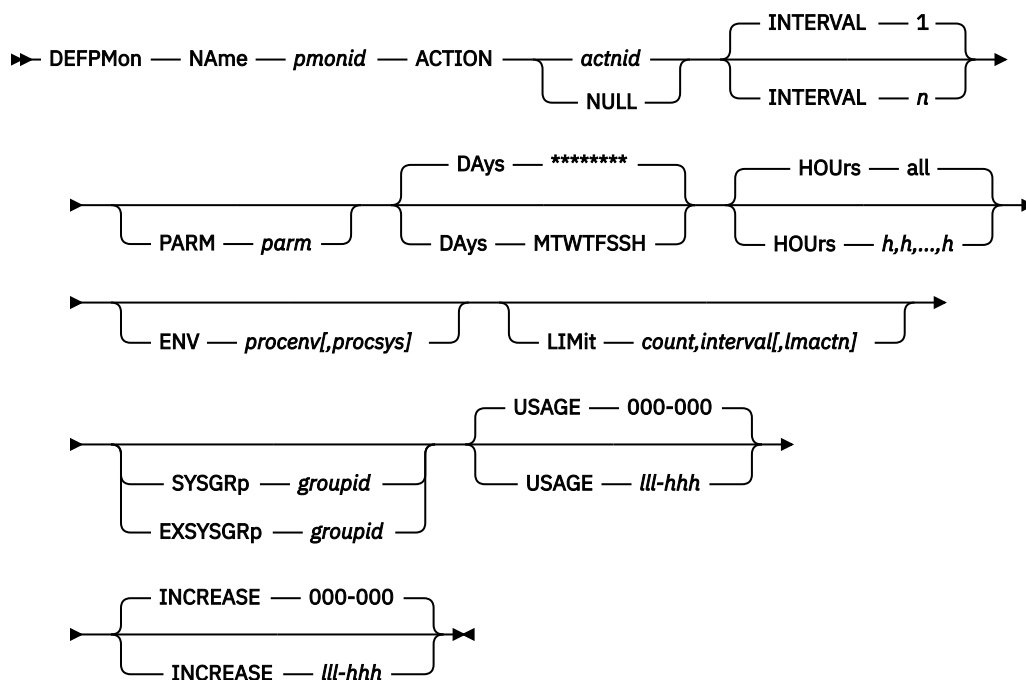


Figure 70. DEFPMON command syntax

## Authorization

The user issuing this command must be authorized for configuration commands.

## Purpose

Use the DEFPMON command to define a page space monitor. A check is performed to determine if the page usage percent or page usage percent increase is within the low and high bounds that are set by the USAGE or INCREASE operand. If so, the action is scheduled for processing. See the usage notes for more information about the message that is passed to the action.

## Operands

### **NAME** *pmonid*

Specifies the name of the page space monitor.

### **ACTION** *actnid*

Specifies the name of the action to process when the page usage falls within the limits for the USAGE or INCREASE operand. This action must have been previously defined using the DEFACTN command or the built-in action NULL.

An action with the name "NULL" is included in Operations Manager. The NULL action is equivalent to specifying DEFACTN NAME NULL and indicates that no action is to be performed. This is useful for rules and monitors that require no action initially, but may require action due to a LIMIT being reached or an idle monitor (via DEFIMON) being triggered.

### **INTERVAL** *n*

Specifies how often to check the page usage for this monitor in minutes. The default is 1 minute.

### **PARM** *parm*

Specifies the parameter or list of parameters that is passed to the ACTION when it is scheduled to run. The value(s) may be entered as *parm*, but must be enclosed in quotation marks (' *parm* ') if there is more than one blank-delimited value.

The DEFACTN command can use the parameter through the &p substitution variable. The parameter is also available to any chained action routine that is defined in the NEXTACTN operand on DEFACTN.

### **DAYS** *MTWTFSSH*

Specifies the days of the week (Monday-Sunday) and if the monitor is active on holidays. For the monitor to be active on a day of the week and holiday, the corresponding location must contain the asterisk (\*) symbol. The default is all days and holidays (\*\*\*\*\*). Refer to [“DEFDATE” on page 89](#) to define holidays.

### **HOURS** *h,h,...,h*

Specifies the hours of the day the monitor is active. Any of the values 0 to 23 may appear in the comma-separated list. Up to 24 values may be present. The default is all hours.

### **ENV** *procenv [,procsys]*

Specifies that the processing environment and the processing system of the action is to be overridden with the supplied values for *procenv* and *procsys*. See [“DEFACTN” on page 84](#) for additional information on overriding the processing environment and processing system.

### **LIMIT** *count,interval[,lmtactn]*

Specifies that the processing of the monitor's ACTION is limited to *count* times within *interval* period of time. Optionally, this command processes the alternate action specified by *lmtactn* if the limit condition is encountered.

- The value for *count* must be from 0 to 999999.
- When specified as a single positive integer, the value for *interval* must be from 0 to 999999 seconds. You can also use the format of ddd-hh:mm:ss, which specifies the interval value in days (ddd), hours (hh), minutes (mm), and seconds (ss). The maximum value for this input format is 999-23:59:59.

For example, a LIMIT of 4 times in 10 seconds could be specified as:

```
LIMIT '4,10'    OR    LIMIT '4,000-00:00:10'
```

- A value of 0 for *count* and *interval* indicates that no processing limit should be applied. If count or interval are 0, both count and interval must be 0, and the *lmtactn* value omitted. If count and interval are non-zero, the alternate action *lmtactn* may be specified. The action *lmtactn* is only processed once if the limit condition is encountered within an interval.

**Note:** If the LIMIT operand is omitted, the default count and interval are obtained from the DEFOPTN command, EMONLIM operand.

**SYSGRP *groupid***

Specifies that this monitor is only applicable for systems where the local system name is in the group named *groupid*.

**EXSYSGRP *groupid***

Specifies that this monitor is only applicable for systems where the local system name is *not* in the group named *groupid*.

**USAGE *lll-hhh***

Specifies the low and high usage percent (000-100) limits to process the action specified in the ACTION operand. If USAGE is not specified, no usage percent check is performed. Both low and high limits must be three digits. The low and high limits are percentages of the total available page space.

**INCREASE *lll-hhh***

Specifies the low and high limits (000-100) for the usage percent increase to process the action specified in the ACTION operand. If INCREASE is not specified, no usage percent increase check is performed. The usage percent increase is computed by subtracting the previous page usage from the current page usage. The low and high limits must be three digits.

**Related reference**

[DEFGROUP](#)

[DEFACTN](#)

[DEFDATE](#)

## Operand value lengths

Table 61 on page 107 lists the operand value lengths for the DEFPMON command.

Table 61. DEFPMON operand value lengths	
Operand	Length
ACTION	1-8
DAYS	8
ENV	1-8
EXSYSGRP	1-8
HOURS	1-72
INCREASE	7
INTERVAL	1-6
LIMIT	3-28
NAME	1-8
PARM	1-120
SYSGRP	1-8
USAGE	7

## Example

An example is provided to help you understand how to use the DEFPMON command, and shows the typical results of this command.

### Example 1: Use the DEFPMON command to define a page space monitor, PSPACE, and an action, PMONIT, to be performed.

When defining the PSPACE page space monitor, a check is performed to determine if the page usage percent is within the low and high bounds that are set by the USAGE operand. For this example, the lower limit of the usage command is 000 and the upper limit is 002. When this condition is met, an action, PMONIT, is scheduled for processing, which displays a message on the TS3163 user console.

```
GOMCMD OPMGRM1 DEFACTN NAME PMONIT COMMAND 'CP MSG TS3163 PAGE SPACE IS &4% FULL' ENV
LVM
Ready; T=0.01/0.01 05:11:16
GOMCMD OPMGRM1 DEFPMON NAME PSPACE ACTION PMONIT USAGE 000-002
Ready; T=0.01/0.01 05:11:27
05:11:30 * MSG FROM OPMGRM1 : PAGE SPACE IS 0% FULL
05:12:00 * MSG FROM OPMGRM1 : PAGE SPACE IS 0% FULL
05:13:00 * MSG FROM OPMGRM1 : PAGE SPACE IS 0% FULL
05:14:00 * MSG FROM OPMGRM1 : PAGE SPACE IS 0% FULL
```

Figure 71. Example of DEFPMON command with DEFACTN command

## Usage notes

1. The VM user ID that is running GOMMAIN (OPMGRM1 by default), must have PRIV D to operate correctly. For more information on privilege classes, refer to [Chapter 2, “Configuring Operations Manager,” on page 5](#).
2. Either USAGE or INCREASE must be specified.
3. The low and high limit checks are inclusive.
4. Monitor checking is performed in the following order:
  - a. USAGE is checked first. If usage falls within the low and high limits, the ACTION is scheduled for USAGE condition and no further checks are made.
  - b. INCREASE is checked second. If usage increase occurs within the low and high limits, the ACTION is scheduled for INCREASE condition.
5. The message field that is passed to the action contains the following fields:
  - &0 — Constant "PAGE"
  - &1 — Name of the page space monitor
  - &2 — Name of the page space monitor's action
  - &3 — Condition field that is passed to the action (USAGE or INCREASE)
  - &4 — Page usage percent
  - &5 — Page usage percent increase for this monitor
  - &6 — Interval of the page monitor

For more information on action substitution variables, see [Appendix C, “Using substitution variables in commands,” on page 183](#).
6. The value for space utilization percentage is obtained by rounding downward. This value may differ from the space utilization percentage computed from other sources.
7. The SYSGRP operand and the EXSYSGRP operand are mutually exclusive.
8. The message field that is passed to the *lmtactn* action of the LIMIT operand contains the following fields:
  - &0 — Identifies the constant LIMIT.
  - &1 — Identifies the constant PAGE.

- &2 — Indicates the name of the page monitor.
- &3 — Indicates the LIMIT count.
- &4 — Indicates the LIMIT interval.
- &5 — Identifies the total number of times that the normal action was suppressed.
- &6-&n — Specifies the message field passed to the previous processing ACTION.

For more information on these substitution variables, see [Appendix C, “Using substitution variables in commands,”](#) on page 183.

## Return codes

Table 62 on page 109 lists the return codes for the DEFPMON command.

Table 62. DEFPMON return codes	
Return code	Description
0	Finished correctly
-100	Error

## DEFRULE

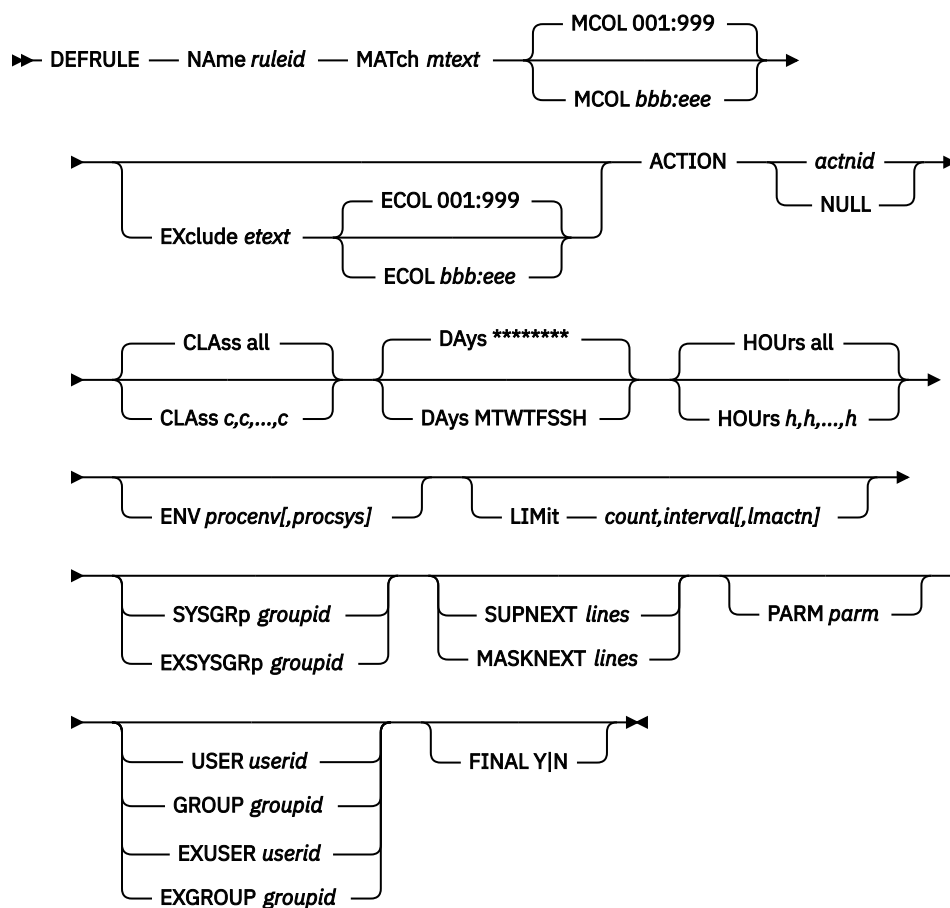


Figure 72. DEFROLE command syntax

## Authorization

The user issuing this command must be authorized for configuration commands.

## Purpose

Use the DEFRULE command to setup a rule that defines a pattern that is to be used for comparing messages from monitored consoles and determining if there is a match. The DEFRULE command also defines an action that is associated with the rule. If conditions for the rule are met, the associated action is scheduled for processing.

## Operands

### NAME *ruleid*

Specifies the name of the rule.

### MATCH *mtext*

Specifies the pattern that will be used to compare against a message to determine if there is a match. Wildcard pattern matching is allowed. If a match is found, the associated action is scheduled for processing. The value must be entered as '*mtext*' if the value contains more than one blank-delimited word.

### MCOL *bbb:eee*

Specifies the beginning and ending columns to search for the match pattern (inclusive). The valid format is *bbb:eee*, where *bbb* is the beginning column and *eee* is the ending column. The default is 001:999. Both the beginning and ending column values must be specified as three digits.

### EXCLUDE *etext*

Specifies the pattern that will be used to compare against a message to determine if there is an exclusion. Wildcard pattern matching is allowed. The value must be entered as '*etext*' if the value contains more than one blank-delimited word. If an exclusion is found, the associated action is *not* scheduled for processing.

### ECOL *bbb:eee*

Specifies the beginning and ending columns to search for the exclude pattern (inclusive). The format is *bbb:eee*, where *bbb* is the beginning column and *eee* is the ending column. The default is 001:999. Both the beginning and ending column values must be specified as three digits.

**Note:** The ECOL operand may be specified only if the EXCLUDE operand is specified.

### ACTION *actnid*

Specifies the name of the associated action. This action must have been previously defined using the DEFACTN command or the built-in action NULL.

An action with the name "NULL" is included in Operations Manager. The NULL action is equivalent to specifying DEFACTN NAME NULL and indicates that no action is to be performed. This is useful for rules and monitors that require no action initially, but may require action due to a LIMIT being reached or an idle monitor (via DEFIMON) being triggered.

### CLASS *c,c,...,c*

Specifies that this rule is only applicable for the listed message classes. Any value from 1 to 255 may appear in the comma-separated list. Up to 32 values may be present. The default is all classes. For a list of valid message classes, see [Appendix C, "Using substitution variables in commands," on page 183](#).

### DAYS *MTWTFSSH*

Specifies the days of the week (Monday-Sunday) and if the rule is active on holidays. For the rule to be active on a day of the week and holiday, the corresponding location must contain the asterisk (\*) symbol. The default is all days and holidays (\*\*\*\*\*). See ["DEFDATE" on page 89](#) for information on defining holidays.

### HOURS *h,h,...,h*

Specifies the hours of the day the rule is active. Any of the values from 0 to 23 may appear in the comma-separated list. Up to 24 values may be present. The default is all hours.

### ENV *procenv[,procsys]*

Specifies that the processing environment and system of the action is to be overridden with the values assigned to *procenv* and *procsys*. See ["DEFACTN" on page 84](#) for more information on overriding the processing environment and system, and valid values for *procenv* and *procsys*.

**LIMIT *count, interval[, lmtactn]***

Specifies that the processing of the monitor's ACTION is limited to *count* times within *interval* period of time. Optionally, this command processes the alternate action specified by *lmtactn* if the limit condition is encountered.

- The value for *count* must be from 0 to 999999.
- When specified as a single positive integer, the value for *interval* must be from 0 to 999999 seconds. You can also use the format of *ddd-hh:mm:ss*, which specifies the interval value in days (*ddd*), hours (*hh*), minutes (*mm*), and seconds (*ss*). The maximum value for this input format is 999-23:59:59.

For example, a LIMIT of 4 times in 10 seconds could be specified as:

```
LIMIT '4,10'    OR    LIMIT '4,000-00:00:10'
```

- A value of 0 for *count* and *interval* indicates that no processing limit should be applied. If *count* or *interval* are 0, both *count* and *interval* must be 0, and the *lmtactn* value omitted. If *count* and *interval* are non-zero, the alternate action *lmtactn* may be specified. The action *lmtactn* is only processed once if the limit condition is encountered within an interval.

**Note:** If the LIMIT operand is omitted, the default *count* and *interval* are obtained from the DEFOPTN command, EMONLIM operand.

**SYSGRP *groupid***

Specifies that this rule is only applicable for systems where the local system name is in the group named *groupid*.

**EXSYSGRP *groupid***

Specifies that this rule is only applicable for systems where the local system name is *not* in the group named *groupid*.

**SUPNEXT *lines***

Suppresses the specified number of output lines from each user's console affected by this rule. The suppressed lines begin with the line following the line that triggered this rule. The value for *lines* can be from 0 to 999999. If lines are already suppressed for the affected user's console, the number of remaining suppressed lines will be changed to *lines*. If SUPNEXT is not specified on a triggered rule, the default is not to change the line suppression count for the affected user's console.

**MASKNEXT *lines***

Masks the specified number of lines in the Operations Manager log coming from each user's console affected by this rule. The masked lines begin with the line following the line that triggered this rule. The value for *lines* can be from 0 to 999999. If lines are already masked for the affected user's console, the number of remaining masked lines will be changed to *lines*. If MASKNEXT is not specified on a triggered rule, the default is not to change the line mask count for the affected user's console.

**PARM *parm***

Specifies the parameter or list of parameters that are passed to the ACTION when it is scheduled to run. The value(s) may be entered as *parm*, but must be enclosed in quotation marks (' *parm* ') if there is more than one blank-delimited value. The DEFACTN command can use the parameter through the &p substitution variable. The parameter is also available to any chained action routine that is defined in the NEXTACTN operand on the DEFACTN command.

**USER *userid***

Specifies this rule is only applicable for messages from the specified *userid*. Wildcard characters are not supported.

**GROUP *groupid***

Specifies this rule is only applicable for messages from users in the group named *groupid*.

**EXUSER *userid***

Specifies this rule is only applicable for messages NOT from *userid*. Wildcard characters are not supported.

**EXGROUP *groupid***

Specifies this rule is only applicable for messages NOT from users in the group named *groupid*.

## FINAL Y|N

Specifies if this is the final rule to process for the current message if the ACTION was scheduled. The default value is FINAL N, which indicates that rule processing continues for this message. FINAL Y indicates there will be no further rule processing for this message.

### Related reference

[DEFACTN](#)

[DEFDATE](#)

[DEFGROUP](#)

## Operand value lengths

Table 63 on page 112 lists the operand value lengths for the DEFRULE command.

Table 63. DEFULE operand value lengths	
Operand	Length
ACTION	1-8
DAYS	8
ECOL	7
ENV	1-8
EXCLUDE	1-60
EXGROUP	1-8
EXSYSGRP	1-8
EXUSER	1-8
FINAL	1
GROUP	1-8
HOURS	1-72
LIMIT	3-28
MATCH	1-60
MCOL	7
NAME	1-8
PARM	1-120
SUPNEXT	1-6
SYSGRP	1-8
USER	1-8

## Example

An example is provided to help you understand how to use the DEFRULE command, and shows the typical results of this command.

**Example 1: Use the DEFRULE command to create a rule (SAMPLE2) to test for a match for '\*SESS\*' in messages for user TS3163.**

The example below shows that action RTEST is to be taken when the criteria for DEFRULE are met.



```
GOMCMD OPMGRM1 DEFACTN NAME RTEST COMMAND 'CP MSG TS3163 HELLO WORLD' ENV
LVM
Ready; T=0.01/0.01 04:42:49
GOMCMD OPMGRM1 DEFRULE NAME SAMPLE2 MATCH '*SESS*' ACTION RTEST USER TS3163
Ready; T=0.01/0.01 04:44:43
```

Figure 73. DEFRULE command issued with DEFACTN command

When any text containing the string '\*SESS\*' appears on the console of TS3163, the action RTEST gets triggered and the message (below) is displayed on the TS3163 console.

```
09:50:57 * MSG FROM OPMGRM1 : HELLO WORLD
```

Figure 74. Message displayed on console

**Example 2: Use the DEFRULE command to create a rule (RULE1) to test for a match for '\*MACHINE\*' in a message being written to the Operations Manager log and set the masking count to three to mask the next three lines after the matching line.**

```
GOMCMD OPMGRM1 DEFACTN NAME ACT1 COMMAND 'CP MSG MAINT720 LINE MASKED!!' ENV LVM
Ready; T=0.01/0.01 08:11:30
GOMCMD OPMGRM1 DEFRULE NAME RULE1 MATCH '*MACHINE*' ACTION ACT1 MASKNEXT 3
Ready; T=0.01/0.01 08:11:43
```

The Operations Manager log will contain the following.

```
07/18/22 08:15:18 GOMCMD0216L TS3159 "07/18/22 08:15:18 GOMCMD0902I 0 MACHINE MONITOR(S)
DEFINED" VID=*MSG SRC=MASIUCV CLS=
07/18/22 08:15:18 GOMCMD0216L TS3159 "*****" VID=*MSG SRC=MASC
07/18/22 08:15:18 GOMCMD0216L TS3159 "*****"
VID=*MSG SRC=MASIUCV CLS=8
07/18/22 08:15:18 GOMCMD0216L TS3159 "*****"
VID=*MSG SRC=MASIUCV CLS=8
07/18/22 08:15:18 GOMCMD0216L TS3159 "07/18/22 08:15:18 GOMCMD0905I 0 ACTION(S) SCHEDULED
FOR PROCESSING" VID=*MSG SRC=MASIU
07/18/22 08:15:18 GOMCMD0216L TS3159 "07/18/22 08:15:18 GOMCMD0906I 2 GROUP(S) DEFINED"
VID=*MSG SRC=MASIUCV CLS=8
07/18/22 08:15:18 GOMCMD0216L TS3159 "07/18/22 08:15:18 GOMCMD0907I 0 DATES(S) DEFINED"
VID=*MSG SRC=MASIUCV CLS=8
```

**Example 3: Use the DEFRULE command to create a rule (RULE1) to test for a match for '\*MACHINE\*' in a message being written to the Operations Manager log and set the masking count to three to mask the next three lines. Use DEFACTN INPUT MSK to define an action which will mask the matching line upon triggering the rule.**

```
GOMCMD OPMGRM1 DEFACTN NAME ACT1 COMMAND 'CP MSG MAINT720 LINE MASKED!!' ENV LVM INPUT MSK
Ready; T=0.01/0.01 08:17:26
GOMCMD OPMGRM1 DEFRULE NAME RULE1 MATCH '*MACHINE*' ACTION ACT1 MASKNEXT 3
Ready; T=0.01/0.01 08:17:40
```

The Operations Manager log will contain the following.

```
07/18/22 08:18:22 GOMCMD0216L TS3159
"*****" VID=*MSG SRC=MASIUCV CLS=
07/18/22 08:18:22 GOMCMD0216L TS3159
"*****" VID=*MSG SRC=MASC
07/18/22 08:18:22 GOMCMD0216L TS3159 "*****"
VID=*MSG SRC=MASIUCV CLS=8
07/18/22 08:18:22 GOMCMD0216L TS3159 "*****"
VID=*MSG SRC=MASIUCV CLS=8
07/18/22 08:18:22 GOMCMD0216L TS3159 "07/18/22 08:18:22 GOMCMD0905I 0 ACTION(S) SCHEDULED
FOR PROCESSING" VID=*MSG SRC=MASIU
07/18/22 08:18:22 GOMCMD0216L TS3159 "07/18/22 08:18:22 GOMCMD0906I 2 GROUP(S) DEFINED"
VID=*MSG SRC=MASIUCV CLS=8
07/18/22 08:18:22 GOMCMD0216L TS3159 "07/18/22 08:18:22 GOMCMD0907I 0 DATES(S) DEFINED"
VID=*MSG SRC=MASIUCV CLS=8
07/18/22 08:18:22 GOMCMD0216L TS3159 "07/18/22 08:18:22 GOMCMD0908I 0 SPOOL MONITOR(S)
DEFINED" VID=*MSG SRC=MASIUCV CLS=8
```

## Usage notes

1. Wildcard patterns can be used with the DEFRULE command. See [Appendix B, “Wildcard pattern matching,” on page 181](#) for an explanation of wildcard pattern matching.
2. USER, GROUP, EXUSER, and EXGROUP are mutually exclusive.
3. The text specified on the MATCH and EXCLUDE operands must exactly match the full text found in the columns specified in MCOL and ECOL respectively. Use wildcard patterns to look for a substring anywhere in the message.

The following example triggers the action TESTACTN if the text "Testing error" appears anywhere in a message:

```
DEFRULE NAME TEST MATCH '*Testing error' ACTION TESTACTN
```

If you do not specify an asterisk at the end of the MATCH operand value, the matching text must end in the (second) column specified in the MCOL operand. Similarly, if you do not specify an asterisk at the beginning of the MATCH operand value, the matching text must begin in the (first) column specified in the MCOL operand.

The example below triggers the action TESTACTN if the text "Testing error" appears in columns 28 through 40 of a message. The matching text must end in column 40 because there is no asterisk at the end of the matching text. Since the matching text is 13 characters, the beginning column must be 28 even though the MCOL beginning column is defined as 001.

```
DEFRULE NAME TEST MATCH '*Testing error' MCOL 001:040 ACTION TESTACTN
```

4. Notes about the SUPNEXT operand:
  - a. SUPNEXT on a DEFRULE command and INPUT SUP on a DEFACTN command are independent. SUPNEXT suppresses the number of lines specified, starting with the line following the line that triggered the rule. INPUT SUP suppresses the current line triggered by the rule.
  - b. All suppressed lines continue through rule processing.
  - c. Separate values are maintained for SUPNEXT for each console. For example, a rule containing SUPNEXT 10 can affect three consoles. For one of those consoles, the third suppressed line can trigger a different rule that uses SUPNEXT 0 to cancel line suppression for that console. The remaining two consoles still suppress the remaining seven lines.
5. Notes about the MASKNEXT operand:
  - a. MASKNEXT and SUPNEXT are mutually exclusive. If you want to suppress the line from VIEWCON and mask the line in the Operations Manager log, specify separate rules: one with SUPNEXT and one with MASKNEXT.
  - b. MASKNEXT indicates that this rule will only apply to the Operations Manager log and will be ignored for consoles visible via VIEWCON.
  - c. INPUT MSK on a DEFACTN command is dependent on MASKNEXT on DEFRULE. INPUT MSK will only mask the current line if the rule which triggered this action has the MASKNEXT option specified.
  - d. MASKNEXT masks the number of lines specified, starting with the line following the line that triggered the rule. INPUT MSK masks the current line triggered by the rule.
  - e. All masked lines continue through rule processing. Separate values are maintained for MASKNEXT for each monitored console. For example, a rule containing MASKNEXT 10 can affect three monitored consoles. For one of those monitored consoles, the third masked line can trigger a different rule that uses MASKNEXT 0 to cancel line masking for that monitored console. The remaining two monitored consoles still mask the remaining seven lines.
6. The SYSGRP operand and the EXSYSGRP operand are mutually exclusive.
7. The message field that is passed to the *lmtactn* action of the LIMIT operand contains the following fields:
  - &0 — Identifies the constant LIMIT.

- &1 — Identifies the constant RULE.
- &2 — Indicates the name of the rule.
- &3 — Indicates the LIMIT count.
- &4 — Indicates the LIMIT interval.
- &5 — Identifies the total number of times that the normal action was suppressed.
- &6-&n — Specifies the message field passed to the previous processing ACTION.

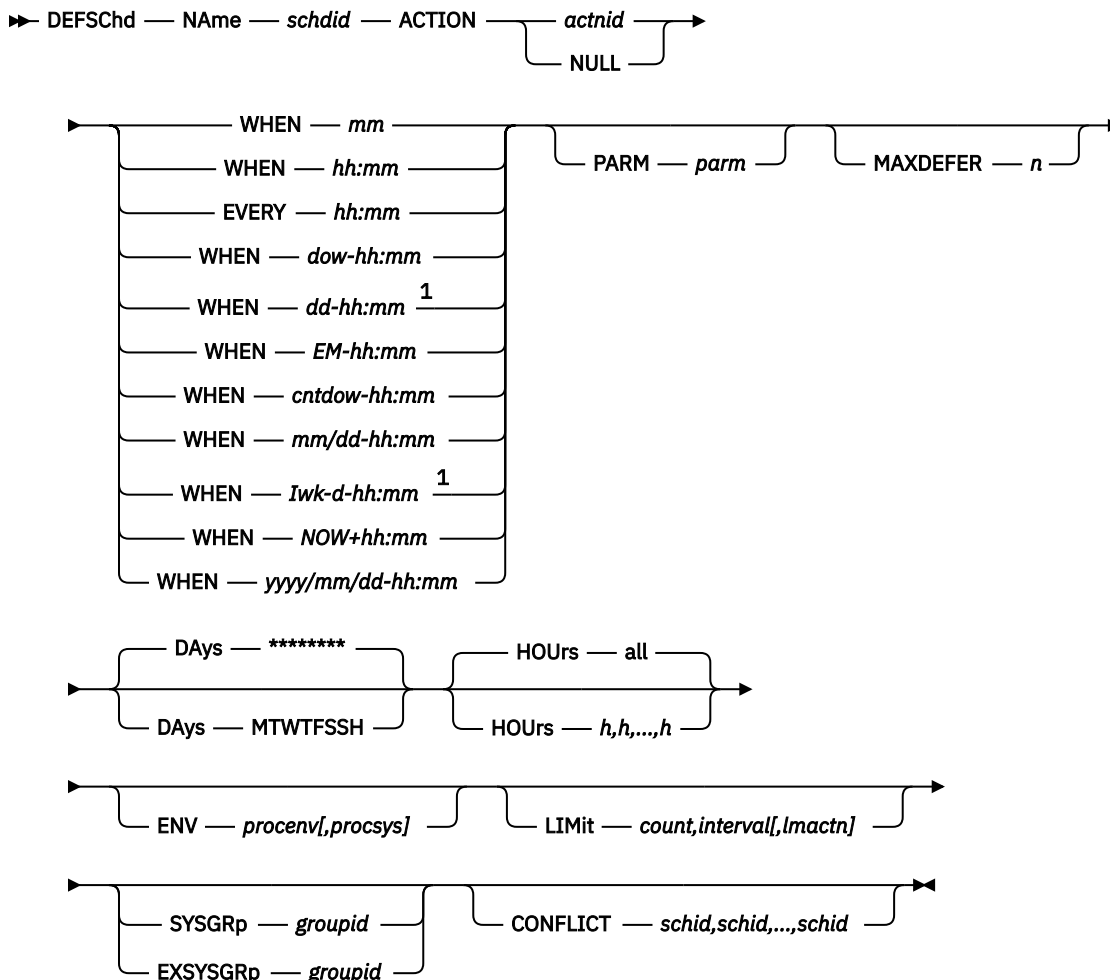
For more information on these substitution variables, see [Appendix C, “Using substitution variables in commands,”](#) on page 183.

## Return Codes

Table 64 on page 115 lists the return codes for the DEFRULE command.

Table 64. DEFRULE return codes	
Return code	Description
0	Finished correctly
-100	Error

## DEFSCHD



Notes:

<sup>1</sup> If the specified value does not occur within a month or year, the execution time is postponed until the value is reached. For example: `WHEN 31-00:00` causes the execution to occur only in months with 31 days. `WHEN 02/29-00:00` causes the execution to occur only in leap years.

Figure 75. DEFSCHD command syntax

## Authorization

The user issuing this command must be authorized for configuration commands.

## Purpose

Use the DEFSCHD command to define a schedule and associate an action with the schedule. The schedule indicates whether the action is processed once, hourly or on a specific hour, daily or on a specific day of the week, on a holiday, monthly, or yearly. It also indicates which days, hours, and holidays are used for processing.

## Operands

### **NAME *schdid***

Specifies the name of the schedule.

### **ACTION *actnid***

Specifies the name of the action to associate with the schedule. This action must have been previously defined using the DEFACTN command or the built-in action NULL.

An action with the name "NULL" is included in Operations Manager. The NULL action is equivalent to specifying `DEFACTN NAME NULL` and indicates that no action is to be performed. This is useful for rules and monitors that require no action initially, but may require action due to a LIMIT being reached or an idle monitor (via DEFIMON) being triggered.

### **WHEN *mm***

Specifies the processing time is hourly. The action is scheduled at *mm* minutes of the hour.

### **WHEN *hh:mm***

Specifies the processing time is daily. The action is scheduled at *hh:mm* time of the day. For example, 01:30 refers to 1:30 A.M., and 13:30 refers to 1:30 P.M.

### **EVERY *hh:mm***

Specifies the processing time is based on an interval. The interval is every *hh:mm* of the day starting with Operations Manager startup, configuration file reload, or DEFSCHD definition.

### **WHEN *dow-hh:mm***

Specifies the processing time is weekly. The action is scheduled on a specific "*dow*" or day of the week (MON, TUE, WED, THU, FRI, SAT, or SUN) at *hh:mm* time of the day.

### **WHEN *dd-hh:mm***

Specifies the processing time is monthly. The action is scheduled on *dd* day of the month at *hh:mm* time of the day.

### **WHEN *EM-hh:mm***

Specifies the processing time is monthly. The action is scheduled on the last day of the month at *hh:mm* time of the day.

### **WHEN *cntdow-hh:mm***

Specifies the processing time is monthly. The action is scheduled on the specified day at *hh:mm* time of the day.

- The *cnt* can be 1ST, 2ND, 3RD, 4TH, 5TH, or LST.
- The *dow* can be MON, TUES, WED, THU, FRI, SAT, SUN, WKD (week day), or WKE (weekend day).

### **WHEN *mm/dd-hh:mm***

Specifies the processing time is yearly. The action is scheduled on *mm/dd* of the year at *hh:mm* time of the day.

**WHEN *Iwk-d-hh:mm***

Specifies the processing time is based upon the ISO week number.

- The *wk* value can be a week number (01-53), the first week (FW), last week (LW), even-numbered week (EV), or odd-numbered week (OD).
- The *d* value is the ISO "day of the week" number (1-7). The days are numbered from 1 (Monday) through 7 (Sunday).
- The action is scheduled at *hh:mm* time of the day.

**WHEN *NOW+hh:mm***

Specifies the processing time to be once, after a delay of *hh:mm* from now. The *hh:mm* may range from +00:00 to +23:59. If omitted, the default is +00:00.

**WHEN *yyyy/mm/dd-hh:mm***

Specifies the processing time to be once on the specified date and time. The action is not scheduled if the date entered has already occurred or never occurs.

**PARM *parm***

Specifies the parameter or list of parameters to be passed to the ACTION when it is scheduled to run. The value(s) may be entered as *parm*, but must be enclosed in quotation marks ('*parm*') if there is more than one blank-delimited value. The DEFACTN command can use the parameter through the &p substitution variable. The parameter is also available to any chained action routine that is defined in the NEXTACTN operand on the DEFACTN command.

**DAYS *MTWTFSSH***

Specifies the days of the week (Monday-Sunday) and if the schedule is active on holidays. For the schedule to be active on a day of the week and holiday, the corresponding location must contain the asterisk (\*) symbol. The default is all days and holidays (\*\*\*\*\*). See [“DEFDATE” on page 89](#) for information on defining holidays.

**HOURS *h,h,...,h***

Specifies the hours of the day the schedule is active. Any of the values from 0 to 23 may appear in the comma-separated list. Up to 24 values may be present. The default is all hours.

**ENV *procenv[,procsys]***

Specifies that the processing environment and system of the action is to be overridden with the values assigned to *procenv* and *procsys*. See [“DEFACTN” on page 84](#) for more information on overriding the processing environment and system, and valid values for *procenv* and *procsys*.

**LIMIT *count,interval[,lmactn]***

Specifies that the processing of the schedule's ACTION is limited to *count* times within *interval* period of time. Optionally, this command processes the alternate action specified by *lmactn* if the limit condition is encountered.

- The value for *count* must be from 0 to 999999.
- When specified as a single positive integer, the value for *interval* must be from 0 to 999999 seconds. You can also use the format of *ddd-hh:mm:ss*, which specifies the interval value in days (*ddd*), hours (*hh*), minutes (*mm*), and seconds (*ss*). The maximum value for this input format is 999-23:59:59.

For example, a LIMIT of 4 times in 10 seconds could be specified as:

```
LIMIT '4,10'    OR    LIMIT '4,000-00:00:10'
```

- A value of 0 for *count* and *interval* indicates that no processing limit should be applied. If count or interval are 0, both count and interval must be 0, and the *lmactn* value omitted. If count and interval are non-zero, the alternate action *lmactn* may be specified. The action *lmactn* is only processed once if the limit condition is encountered within an interval.

**Note:** If the LIMIT operand is omitted, the default count and interval are obtained from the DEFOPTN command, EMONLIM operand.

**SYSGRP *groupid***

Specifies that this schedule is only applicable for systems where the local system name is in the group named *groupid*.

**EXSYSGRP *groupid***

Specifies that this schedule is only applicable for systems where the local system name is *not* in the group named *groupid*.

**MAXDEFER *n***

Specifies the number of minutes the schedule may be delayed from the beginning of the processing due to CONFLICT. If a schedule is not able to begin before the MAXDEFER value is reached, the schedule is advanced to the next processing time. The default is 0 minutes.

**CONFLICT *schid,schid,...,schid***

Specifies up to 16 conflicting schedules. If any of the specified schedules are processing, this schedule is deferred from processing. The schedule is then checked every minute until MAXDEFER is reached.

**Related reference**

[DEFACTN](#)

[DEFDATE](#)

## Operand value lengths

Table 65 on page 118 lists the operand value lengths for the DEFSCHD command.

Table 65. DEFSCHD operand value lengths	
Operand	Length
ACTION	1-8
CONFLICT	1-120
DAYS	8
ENV	1-8
EVERY	5
EXSYSGRP	1-8
HOURS	1-72
LIMIT	3-28
MAXDEFER	1-6
NAME	1-8
PARM	1-120
SYSGRP	1-8
WHEN	2-16

## Examples

An example is provided to help you understand how to use the DEFSCHD command, and shows the typical results of this command.

### Example 1: Use the DEFSCHD command to define a schedule (TESTSCHD) to process an action (PILOT) every minute.

```
GOMCMD OPMGRM1 DEFACTN NAME PILOT COMMAND 'CP MSG TS3163 GOOD MORNING!!!' ENV LVM
Ready; T=0.01/0.01 05:07:52
GOMCMD OPMGRM1 DEFSCHD NAME TESTSCHD ACTION PILOT EVERY 00:01
Ready; T=0.01/0.01 05:19:04
05:19:30 * MSG FROM OPMGRM1 : GOOD MORNING!!!
05:20:00 * MSG FROM OPMGRM1 : GOOD MORNING!!!
05:21:00 * MSG FROM OPMGRM1 : GOOD MORNING!!!
05:22:00 * MSG FROM OPMGRM1 : GOOD MORNING!!!
05:23:00 * MSG FROM OPMGRM1 : GOOD MORNING!!!
```

Figure 76. DEFSCHD command issued

## Usage notes

1. Either the WHEN operand or the EVERY operand must be specified.
2. Any changes made to the hardware TOD clock (which is different than the time zone) do not automatically adjust existing schedules. Stop Operations Manager before making changes to the TOD clock.
3. CONFLICT describes a one-way conflict. If schedules A and B should not process at the same time, schedule A must name schedule B as a conflict *and* schedule B must name schedule A as a conflict.
4. The processing time for any schedule that specifies *mm* for minutes is not necessarily at exactly *mm:00*. The processing begins within 30 seconds of the start of the minute specified.
5. The SYSGRP operand and the EXSYSGRP operand are mutually exclusive.
6. The message field passed to the action contains the following fields:
  - &C — 0
  - &U — Identifies the constant \_GOMSCHD.
  - &0 — Identifies the constant SCHEDULE.
  - &1 — Identifies the name of the schedule.
  - &2 — Identifies the action of the schedule.
  - &3-&99 — Null
7. The message field passed to the *lmtactn* action of the LIMIT operand contains the following fields:
  - &0 — Identifies the constant LIMIT.
  - &1 — Identifies the constant SCHEDULE.
  - &2 — Indicates the name of the schedule.
  - &3 — Indicates the LIMIT count.
  - &4 — Indicates the LIMIT interval.
  - &5 — Identifies the total number of times that the normal action was suppressed.
  - &6-&n — Specifies the message field passed to the previous processing ACTION.

For more information on action substitution variables, see [Appendix C, “Using substitution variables in commands,”](#) on page 183.

## Return codes

[Table 66 on page 120](#) lists the return codes for the DEFSCHD command.

Table 66. DEFSCHD return codes	
Return code	Description
0	Finished correctly
-100	Error

## DEFMON

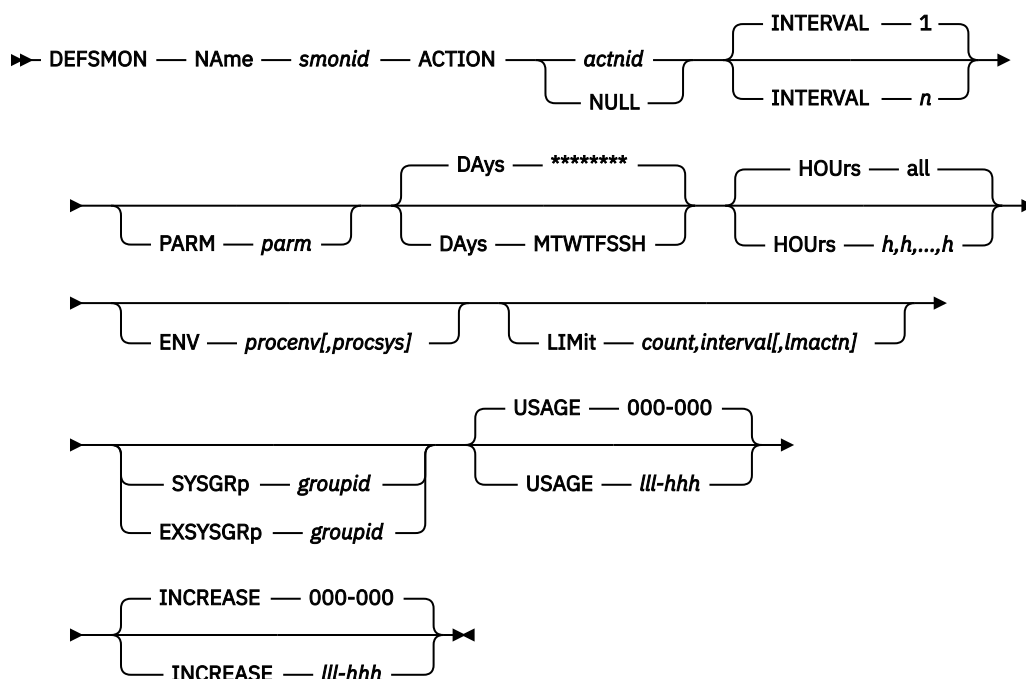


Figure 77. DEFMON command syntax

## Authorization

The user issuing this command must be authorized for configuration commands.

## Purpose

Issue the DEFMON command to define a spool monitor. A check is performed to determine if the spool usage percent or spool usage percent increase is within the low and high bounds set by USAGE or INCREASE. If so, the specified action is scheduled for processing. See [“Usage notes” on page 123](#) for details of the message passed to the action.

## Operands

### NAME *smonid*

Specifies the name of the spool monitor.

### ACTION *actnid*

Specifies the name of the action to process when spool usage falls within the bounds for USAGE or INCREASE. This action must have been previously defined using the DEFACTN command or the built-in action NULL.

An action with the name "NULL" is included in Operations Manager. The NULL action is equivalent to specifying DEFACTN NAME NULL and indicates that no action is to be performed. This is useful for rules and monitors that require no action initially, but may require action due to a LIMIT being reached or an idle monitor (via DEFIMON) being triggered.



**INTERVAL *n***

Specifies how often, in minutes, to check the spool usage for this monitor. The default is 1 minute.

**PARM *parm***

Specifies the parameter or list of parameters that are passed to the ACTION when it is scheduled to run. The value(s) may be entered as *parm*, but must be enclosed in quotation marks (' *parm* ') if there is more than one blank-delimited value. The DEFACTN command can use the parameter through the &p substitution variable. The parameter is also available to any chained action routine that is defined in the NEXTACTN operand on the DEFACTN command.

**DAYS MTWTFSSH**

Specifies the days of the week (Monday-Sunday) and if the monitor is active on holidays. For the monitor to be active on a day of the week and holiday, the corresponding location must contain the asterisk (\*) symbol. The default is all days and holidays (\*\*\*\*\*). See “DEFDATE” on page 89 for information on defining holidays.

**HOURS *h,h,...,h***

Specifies the hours of the day the monitor is active. Any of the values from 0 to 23 may appear in the comma-separated list. Up to 24 values may be present. The default is all hours.

**ENV *procenv[,procsys]***

Specifies that the processing environment and system of the action is to be overridden with the values assigned to *procenv* and *procsys*. See “DEFACTN” on page 84 for more information on overriding the processing environment and system, and valid values for *procenv* and *procsys*.

**LIMIT *count,interval[,lmtactn]***

Specifies that the processing of the monitor's ACTION is limited to *count* times within *interval* period of time. Optionally, this command processes the alternate action specified by *lmtactn* if the limit condition is encountered.

- The value for *count* must be from 0 to 999999.
- When specified as a single positive integer, the value for *interval* must be from 0 to 999999 seconds. You can also use the format of ddd-hh:mm:ss, which specifies the interval value in days (ddd), hours (hh), minutes (mm), and seconds (ss). The maximum value for this input format is 999-23:59:59.

For example, a LIMIT of 4 times in 10 seconds could be specified as:

```
LIMIT '4,10'    OR    LIMIT '4,000-00:00:10'
```

- A value of 0 for *count* and *interval* indicates that no processing limit should be applied. If count or interval are 0, both count and interval must be 0, and the *lmtactn* value omitted. If count and interval are non-zero, the alternate action *lmtactn* may be specified. The action *lmtactn* is only processed once if the limit condition is encountered within an interval.

**Note:** If the LIMIT operand is omitted, the default count and interval are obtained from the DEFOPTN command, EMONLIM operand.

**SYSGRP *groupid***

Specifies that this monitor is only applicable for systems where the local system name is in the group named *groupid*.

**EXSYSGRP *groupid***

Specifies that this monitor is only applicable for systems where the local system name is *not* in the group named *groupid*.

**USAGE *lll-hhh***

Specifies the low and high usage percent limits (000-100) to process the ACTION. If USAGE is not specified, no usage percent check is performed. Both low and high limits must be three digits. The low and high limits are percentages of the total available spool space.

**INCREASE *lll-hhh***

Specifies the low and high limits for the usage percent increase (000-100) to process the ACTION. If INCREASE is not specified, no usage percent increase check is performed. The usage percent increase is computed by subtracting the previous spool usage percent (at the previous interval) from

the current spool usage percent. The low and high limits must be three digits. These limits are percentages of the total available spooling space.

#### Related reference

[DEFACTN](#)

[DEFDATE](#)

## Operand value lengths

Table 67 on page 122 lists the operand value lengths for the DEFMON command.

Table 67. DEFMON operand value lengths	
Operand	Length
ACTION	1-8
DAYS	8
ENV	1-8
EXSYSGRP	1-8
HOURS	1-72
INCREASE	7
INTERVAL	1-6
LIMIT	3-28
NAME	1-8
PARM	1-120
SYSGRP	1-8
USAGE	7

## Example

An example is provided to help you understand how to use the DEFMON command, and shows the typical results of this command.

**Example 1: Use the DEFMON command to define a spool monitor, SMON, that checks the spool USAGE range every minute (since no INTERVAL is specified). If the spool usage is between 0% - 2% (inclusive), a message is sent to TS3163.**

```
GOMCMD OPMGRM1 DEFMON NAME SMON ACTION ASPOOL USAGE 000-002
Ready; T=0.01/0.01 07:03:30

GOMCMD OPMGRM1 DEFACTN NAME ASPOOL COMMAND'CP MSG TS3163 SAYS SPOOL IS &4% FULL' ENV
LVM
Ready; T=0.01/0.01 07:03:01

RESULT:-
07:07:00 * MSG FROM OPMGRM1 : SAYS SPOOL IS 1% FULL
07:07:30 * MSG FROM OPMGRM1 : SAYS SPOOL IS 1% FULL
07:08:30 * MSG FROM OPMGRM1 : SAYS SPOOL IS 1% FULL
07:09:30 * MSG FROM OPMGRM1 : SAYS SPOOL IS 1% FULL
07:10:30 * MSG FROM OPMGRM1 : SAYS SPOOL IS 1% FULL
```

Figure 78. Example of DEFMON command with DEFACTN command

## Usage notes

1. The VM user ID running GOMMAIN must have PRIV D to operate correctly. For more information on privilege classes, refer to [Chapter 2, “Configuring Operations Manager,” on page 5](#).
2. Either USAGE or INCREASE must be specified.
3. The low and high bound checks are inclusive.
4. Monitor checking is performed in the following order:
  - a. USAGE is checked first. If usage falls within the low and high limits, the ACTION is scheduled for USAGE condition and no further checks are made.
  - b. INCREASE is checked second. If a usage increase occurs within the low and high limits, the ACTION is scheduled for INCREASE condition.
5. The message field passed to the action contains the following fields:
  - &0 – Constant "SPOOL".
  - &1 – The name of the spool monitor.
  - &2 – The action of the spool monitor.
  - &3 – The condition field passed to the action. The valid conditions are USAGE or INCREASE.
  - &4 – The spool usage percentage.
  - &5 – The spool usage percent increase for this monitor.
  - &6 – The spool file usage percentage.
  - &7 – The spool file usage percentage increase for this monitor.
  - &8 – The interval of the spool monitor.

For more information on action substitution variables, see [Appendix C, “Using substitution variables in commands,” on page 183](#).
6. The value for space utilization percentage is obtained by rounding downward. This value may differ from the space utilization percentage computed from other sources.
7. The SYSGRP operand and the EXSYSGRP operand are mutually exclusive.
8. The message field passed to the action associated with the Limit operand value, "lmtactn" contains the following fields:
  - &0 – Identifies the constant LIMIT.
  - &1 – Identifies the constant SPOOL.
  - &2 – Indicates the name of the spool monitor.
  - &3 – Indicates the LIMIT count.
  - &4 – Indicates the LIMIT interval.
  - &5 – Identifies the total number of times that the normal action was suppressed.
  - &6-&n – Specifies the message field passed to the previous processing ACTION.

## Return codes

[Table 68 on page 123](#) lists the return codes for the DEFMON command.

Table 68. DEFMON return codes	
Return code	Description
0	Finished correctly
-100	Error

# DELECTN

➤ DELACTn — NAME — actnid ➤

Figure 79. DELACTN command syntax

## Authorization

The user issuing this command must be authorized for configuration commands.

## Purpose

Issue the DELACTN command to delete an action.

### Related reference

[DEFACTN](#)

## Operands

### NAME *actnid*

Specifies a name for the action.

An action with the name "NULL" is included in Operations Manager. The NULL action is equivalent to specifying DEFACTN NAME NULL and indicates that no action is to be performed. You cannot specify the value "NULL" for the NAME operand because it is already a built-in action.

## Operand value lengths

Table 69 on page 124 lists the operand value lengths for the DELACTN command.

Table 69. DELACTN operand value lengths	
Operand	Length
NAME	1-8

## Example

An example is provided to help you understand how to use the DELACTN command and to show the typical results of this command.

### Example 1: Use the DELACTN command to delete an action, TEST2.

```
GOMCMD OPMGRM1 DELACTN NAME TEST2
Ready; T=0.01/0.01 02:27:09
```

Figure 80. DELACTN command issued

## Usage notes

None

## Return codes

Table 70 on page 124 lists the return codes for the DELACTN command.

Table 70. DELACTN return codes	
Return code	Description
0	Finished correctly

Table 70. DELACTN return codes (continued)	
Return code	Description
-100	Error

## DELDATE

➡ DELDate — NAmE — dateId ➡

Figure 81. DELDATE command syntax

### Authorization

The user issuing this command must be authorized for configuration commands.

### Purpose

Issue the DELDATE command to delete a holiday.

#### Related reference

[DEFDATE](#)

### Operands

**NAmE** *dateId*

Specifies the name of the holiday to delete.

### Operand value lengths

Table 71 on page 125 lists the operand value lengths for the DELDATE command.

Table 71. DELDATE operand value lengths	
Operand	Length
NAME	1-8

### Example

An example is provided to help you understand how to use the DELDATE command, and shows the typical results of this command.

#### Example 1: Use the DELDATE command to delete a previously defined holiday, which is XMAS.

```
GOMCMD OPMGRM1 DELDATE NAME XMAS
Ready; T=0.01/0.01 06:26:27
```

Figure 82. DELDATE command issued

### Usage notes

None

### Return codes

Table 72 on page 126 lists the return codes for the DELDATE command.

Table 72. DELDATE return codes	
Return code	Description
0	Finished correctly
-100	Error

## DELEMON

➡ DELEMon — NAmE — *emonid* ➡

Figure 83. DELEMON command syntax

### Authorization

The user issuing this command must be authorized for configuration commands.

### Purpose

Issue the DELEMON command to delete an event monitor.

#### Related reference

[DEFEMON](#)

### Operands

NAmE *emonid*

Specifies the name of the event monitor to delete.

### Operand value lengths

Table 73 on page 126 lists the operand value lengths for the DELEMON command.

Table 73. DELEMON operand value lengths	
Operand	Length
NAME	1-8

### Example

An example is provided to help you understand how to use the DELEMON command, and shows the typical results of this command.

**Example 1: Use the DELEMON command to delete a previously defined event monitor, EVENTMON.**

```
GOMCMD OPMGRM1 DELEMON NAME EVENTMON
Ready; T=0.01/0.01 06:26:27
```

Figure 84. DELEMON command issued

### Usage notes

None

### Return codes

Table 74 on page 127 lists the return codes for the DELEMON command.

Table 74. DELEMON return codes	
Return code	Description
0	Finished correctly
-100	Error

## DELGROUP

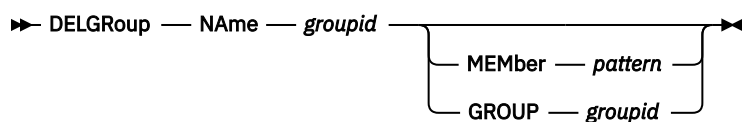


Figure 85. DELGROUP command syntax

## Authorization

The user issuing this command must be authorized for configuration commands.

## Purpose

Issue the DELGROUP command to delete members from a group.

Before deleting a member from a group, keep in mind that several commands and operands may refer to the group. These commands include the DEFEMON, DEFIMON, DEFMMON, DEFPMON, DEFRULE, DEFSCHD, and DEFSMON commands.

### Related reference

[DEFGROUP](#)

## Operands

### NAME *groupid*

Specifies the group name.

### GROUP *groupid*

Specifies a group of members to delete from the group.

### MEMBER *pattern*

Specifies a member to delete from the group. Using the MEMBER operand allows groups to be used as a generalized collection of members and searched for inclusion/exclusion as needed. The pattern may contain wildcard matching information. The wildcard rules are explained in the [Appendix B, “Wildcard pattern matching,”](#) on page 181.

## Operand value lengths

Table 75 on page 127 lists the operand value lengths for the DELGROUP command.

Table 75. DELGROUP operand value lengths	
Operand	Length
MEMBER	1-8
NAME	1-8
USER	1-8

## Example

An example is provided to help you understand how to use the DELGROUP command, and shows the typical results of this command.

### Example 1: Use the DELGROUP command to delete a member, MAINT640, from the RANGERS group.

```
GOMCMD OPMGRM1 DELGROUP NAME RANGERS MEMBER MAINT640
Ready; T=0.01/0.01 06:56:25
```

Figure 86. DELGROUP command issued

## Usage notes

1. The value of the NAME and of the MEMBER (or GROUP) operands must completely match a previously defined value.
2. Either the MEMBER operand or the GROUP operand must be specified.
3. Only the unique combination of NAME and MEMBER (or GROUP) is removed. Other NAME and MEMBER (or GROUP) combinations are not removed.
4. The MEMBER operand and GROUP operand are mutually exclusive.

## Return codes

Table 76 on page 128 lists the return codes for the DELGROUP command.

Table 76. DELGROUP return codes	
Return code	Description
0	Finished correctly
-100	Error

## DELIMON

➡ DELIMon — NAmE — imonid ➡

Figure 87. DELIMON command syntax

## Authorization

The user issuing this command must be authorized for configuration commands.

## Purpose

Issue the DELIMON command to delete an idle monitor.

## Operands

NAmE *imonid*

Specifies the name of the idle monitor.

## Operand value lengths

Table 77 on page 129 lists the operand value lengths for the DELIMON command.



Table 77. DELIMON operand value lengths	
Operand	Length
NAME	1-8

## Example

An example is provided to help you understand how to use the DELIMON command, and shows the typical results of this command.

### Example 1: Use the DELIMON command to delete an idle monitor, IDMON.

```
GOMCMD OPMGRM1 DELIMON NAME IDMON
Ready; T=0.01/0.01 06:26:27
```

Figure 88. DELIMON command issued

## Usage notes

None

## Return codes

Table 78 on page 129 lists the return codes for the DELIMON command.

Table 78. DELIMON return codes	
Return code	Description
0	Finished correctly
-100	Error

## DELMON

►► DELMOn — NAmE — mmonid ◄◄

Figure 89. DELMMON command syntax

## Authorization

The user issuing this command must be authorized for configuration commands.

## Purpose

Issue the DELMMON command to delete a virtual machine monitor.

### Related reference

[DEFMMON](#)

## Operands

**NAmE** *mmonid*

Specifies the name of the machine monitor to delete.

## Operand value lengths

Table 79 on page 130 lists the operand value lengths for the DELMMON command.

Table 79. DELMMON operand value lengths	
Operand	Length
NAME	1-8

## Example

An example is provided to help you understand how to use the DELMMON command, and shows the typical results of this command.

### Example 1: Use the DELMMON command to delete a virtual machine monitor, TEST9MON.

```
GOMCMD OPMGRM1 DELMMON NAME TEST9MON
Ready; T=0.01/0.01 02:17:29
```

Figure 90. DELMMON command

## Usage note

The status of the monitored virtual machine is not changed.

## Return codes

Table 80 on page 130 lists the return codes for the DELMMON command.

Table 80. DELMMON return codes	
Return code	Description
0	Finished correctly
-100	Error

## DELPMON

➤ DELPMon — NAmE — pmonid ➤

Figure 91. DELPMON command syntax

## Authorization

The user issuing this command must be authorized for configuration commands.

## Purpose

Issue the DELPMON command to delete a page space monitor.

## Operands

**NAmE** *pmonid*  
Specifies the name of the page space monitor to delete.

## Operand value lengths

Table 81 on page 131 lists the operand value lengths for the DELPMON command.

Table 81. DELPMON operand value lengths	
Operand	Length
NAME	1-8

## Example

An example is provided to help you understand how to use the DELPMON command, and shows the typical results of this command.

### Example 1: Use the DELPMON command to delete a page space monitor, PSPACE.

```
GOMCMD OPMGRM1 DELPMON NAME PSPACE
Ready; T=0.01/0.01 02:17:29
```

Figure 92. DELPMON command

## Usage notes

None

## Return codes

Table 82 on page 131 lists the return codes for the DELPMON command.

Table 82. DELPMON return codes	
Return code	Description
0	Finished correctly
-100	Error

## DELRULE

►► DELRULE — NAME — ruleid ◄◄

Figure 93. DELRULE command syntax

## Authorization

The user issuing this command must be authorized for configuration commands.

## Purpose

Issue the DELRULE command to delete a rule.

### Related reference

[DEFRULE](#)

## Operands

NAME *ruleid*

Specifies the name of the rule to delete.

## Operand value lengths

Table 83 on page 132 lists the operand value lengths for the DELRULE command.

Table 83. DELRULE operand value lengths	
Operand	Length
NAME	1-8

## Example

An example is provided to help you understand how to use the DELRULE command, and shows the typical results of this command.

### Example 1: Use the DELRULE command to delete a rule, SAMPLE2.

```
GOMCMD OPMGRM1 DELRULE NAME SAMPLE2
Ready; T=0.01/0.01 05:51:48
```

Figure 94. DELRULE command issued

## Usage notes

None

## Return Codes

Table 84 on page 132 lists the return codes for the DELRULE command.

Table 84. DELRULE return codes	
Return code	Description
0	Finished correctly
-100	Error

## DELSCHD

➤ DELSchd — NAmE — schdid ➤

Figure 95. DELSCHD command syntax

## Authorization

The user issuing this command must be authorized for configuration commands.

## Purpose

Issue the DELSCHD command to delete a schedule.

### Related reference

[DEFSCHD](#)

## Operands

**NAmE** *schdid*

Specifies the name of the schedule to delete.

## Operand value lengths

Table 85 on page 133 lists the operand value lengths for the DELSCHD command.

Table 85. DELSCHD operand value lengths	
Operand	Length
NAME	1-8

## Example

An example is provided to help you understand how to use the DELSCHD command, and shows the typical results of this command.

### Example 1: Use the DELSCHD command to delete a schedule, TESTSCHD.

```
GOMCMD OPMGRM1 DELSCHD NAME TESTSCHD
Ready; T=0.01/0.01 05:28:49
```

Figure 96. DELSCHD command

## Usage notes

None

## Return codes

Table 86 on page 133 lists the return codes for the DELSCHD command.

Table 86. DELSCHD return codes	
Return code	Description
0	Finished correctly
-100	Error

## DELSMON

➤ DELSMon — NAmE — smonid ➤

Figure 97. DELSMON command syntax

## Authorization

The user issuing this command must be authorized for configuration commands.

## Purpose

Issue the DELSMON command to delete a spool usage monitor.

## Operands

**NAmE** *smonid*

Specifies the name of the spool usage monitor to delete.

**Related reference**

[DEFMON](#)

## Operand value lengths

Table 87 on page 134 lists the operand value lengths for the DELSMON command.

Table 87. DELSMON operand value lengths	
Operand	Length
NAME	1-8

## Example

An example is provided to help you understand how to use the DELSMON command, and shows the typical results of this command.

**Example 1: Use the DELSMON command to delete a previously defined spool usage monitor, SMON.**

```
GOMCMD OPMGRM1 DELSMON NAME SMON
Ready; T=0.01/0.01 06:26:27
```

Figure 98. DELSMON command issued

## Usage notes

None

## Return codes

Table 88 on page 134 lists the return codes for the DELSMON command.

Table 88. DELSMON return codes	
Return code	Description
0	Finished correctly
-100	Error

## RESUME

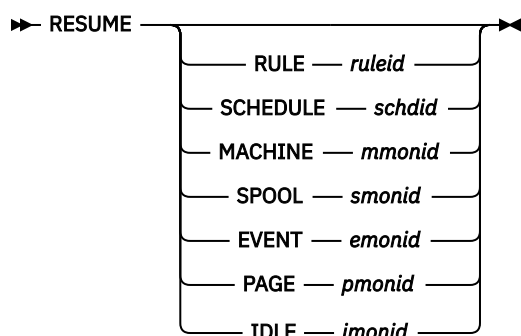


Figure 99. RESUME command syntax

## Authorization

The user issuing this command must be authorized for configuration commands.

## Purpose

Issue the RESUME command to activate a resource (such as a rule, schedule, or monitor) from the suspended state.

The valid resources are:

- RULE
- MACHINE
- SCHEDULE
- SPOOL
- EVENT
- PAGE
- IDLE

## Operands

### **RULE**

Specifies the *ruleid* from DEFRULE.

### **MACHINE**

Specifies the *mmonid* from DEFMMON.

### **SCHEDULE**

Specifies the *schdid* from DEFSCHD.

### **SPOOL**

Specifies the *smonid* from DEFSMON.

### **EVENT**

Specifies the *emonid* from DEFEMON.

### **PAGE**

Specifies the *pmonid* from DEFPMON.

### **IDLE**

Specifies the *imonid* from DEFIMON.

### **Related reference**

[DEFEMON](#)

[DEFIMON](#)

[DEFMMON](#)

[DEFPMON](#)

[DEFRULE](#)

[DEFSCHD](#)

[DEFSMON](#)

[SUSPEND](#)

## Operand value lengths

Table 89 on page 135 lists the operand value lengths for the RESUME command.

Table 89. RESUME operand value lengths	
Operand	Length
EVENT	1-8
IDLE	1-8
MACHINE	1-8
PAGE	1-8
RULE	1-8
SCHEDULE	1-8
SPOOL	1-8

## Example

An example is provided to help you understand how to use the RESUME command, and shows the typical results of this command.

### Example 1: Use the RESUME command to reactivate a spool usage monitor, SMON, from a suspended (SUSPEND) state.

The example below shows the following:

1. Spool monitor, SMON, is defined using the DEFMON command.
2. An action, ASPOOL, is triggered when the USAGE percentage (000-002) is met.
3. The spool monitor, SMON, is SUSPENDED from further processing at 4:15:35.
4. The spool monitor, SMON, is instructed to RESUME processing at 4:24:29.

```
GOMCMD OPMGRM1 DEFACTN NAME ASPOOL COMMAND 'CP MSG TS3163 SAYS SPOOL IS &4% FULL' ENV
LVM
Ready; T=0.01/0.01 04:12:48

GOMCMD OPMGRM1 DEFMON NAME SMON ACTION ASPOOL USAGE 000-002
Ready; T=0.01/0.01 04:12:57

04:13:00 * MSG FROM OPMGRM1 : SAYS SPOOL IS 1% FULL
04:13:30 * MSG FROM OPMGRM1 : SAYS SPOOL IS 1% FULL
04:14:30 * MSG FROM OPMGRM1 : SAYS SPOOL IS 1% FULL
04:15:30 * MSG FROM OPMGRM1 : SAYS SPOOL IS 1% FULL

GOMCMD OPMGRM1 SUSPEND SPOOL SMON
Ready; T=0.01/0.01 04:15:35
GOMCMD OPMGRM1 RESUME SPOOL SMON
Ready; T=0.01/0.01 04:24:29
04:24:30 * MSG FROM OPMGRM1 : SAYS SPOOL IS 1% FULL
04:25:30 * MSG FROM OPMGRM1 : SAYS SPOOL IS 1% FULL
04:26:30 * MSG FROM OPMGRM1 : SAYS SPOOL IS 1% FULL
04:27:30 * MSG FROM OPMGRM1 : SAYS SPOOL IS 1% FULL
04:28:30 * MSG FROM OPMGRM1 : SAYS SPOOL IS 1% FULL
04:29:30 * MSG FROM OPMGRM1 : SAYS SPOOL IS 1% FULL
04:30:30 * MSG FROM OPMGRM1 : SAYS SPOOL IS 1% FULL
04:31:30 * MSG FROM OPMGRM1 : SAYS SPOOL IS 1% FULL
```

Figure 100. RESUME command issued

## Usage notes

1. One of the resources must be specified.
2. Only one of the resources must be specified.
3. The use of wildcard characters is not allowed.
4. Use ALL in place of ruleid, mmonid, schdid, smonid, emonid, pmonid, or imonid to resume all definitions for the specified resource type. For example, the following command resumes all rules that are in the suspended state:

```
RESUME RULE(ALL)
```

## Return codes

Table 90 on page 136 lists the return codes for the RESUME command.

Table 90. RESUME return codes	
Return Code	Description
0	Finished correctly
-100	Error



# RUN

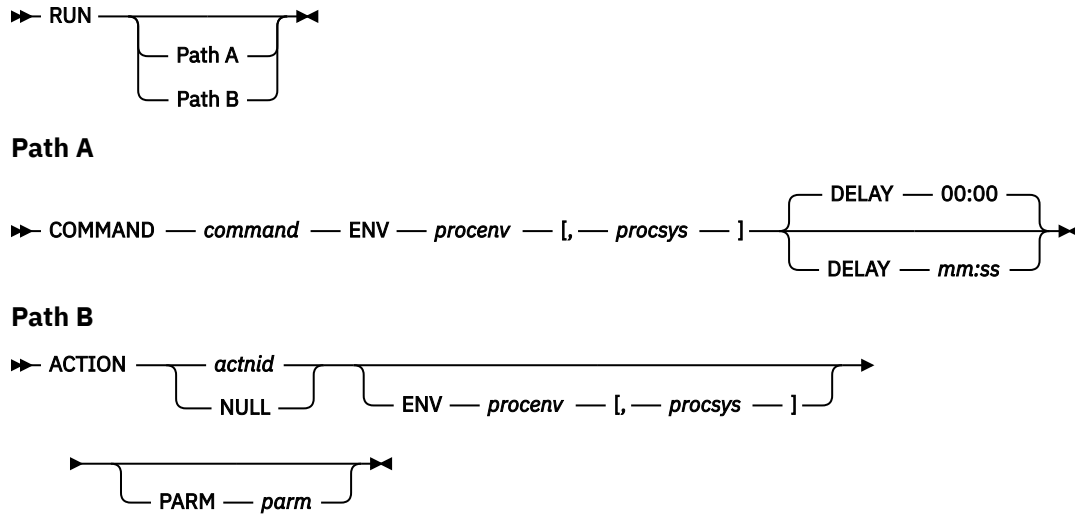


Figure 101. RUN command syntax

## Authorization

The user issuing this command must be authorized for configuration commands.

## Purpose

Issue the RUN command to run a command or action.

## Operands

### Path A options

#### COMMAND *command*

Specifies the command to be processed. This can be any CMS or CP command. The value may be entered as 'command'.

#### ENV *procenv* [, *procsys*]

Specifies the processing environment and processing system where the command or actions must be processed. The valid values for *procenv* and *procsys* are:

##### LVM

Specifies that the command must be processed by the local virtual machine. Use caution when specifying LVM. Any command or module that processes in the LVM preempts processing of Operations manager messages and can possibly affect performance and reliability of Operations Manager.

##### SVM

Specifies that the command must be processed by any available action processing server (service machine).

##### *userid*

Specifies that the command must be processed by only the indicated action processing server.

##### *wgrp*

Specifies that the command must be processed by only the action processing servers participating in the work group.

**sysid**

Specifies that the command must be processed on the system named *sysid*. To specify the local system, you can use an asterisk (\*). The *sysid* should match the local system name, or the value of the NAME operand from a DEFIPCS command with the operand APPL GOMRSCC.

**DELAY mm:ss**

Specifies an amount of time to delay before processing the command. Input is in minutes and seconds.

**Path B options****ACTION actnid**

Specifies the action to be processed.

An action with the name "NULL" is included in Operations Manager. The NULL action is equivalent to specifying DEFACTN NAME NULL and indicates that no action is to be performed. This is useful for rules and monitors that require no action initially, but are being monitored by an idle monitor (via DEFIMON).

**ENV procenv[,procsys]**

Specifies the processing environment and processing system of the action must be overridden with the values provided for *procenv* and *procsys*. See “DEFACTN” on page 84 for additional information on overriding the processing environment and processing system.

**PARM parm**

Specifies the parameter passed to the action when it is scheduled for processing. The value may be entered as 'parm'. See “DEFACTN” on page 84 for a list of parameter values.

**Operand value lengths**

Table 91 on page 138 lists the operand value lengths for the RUN command.

Table 91. RUN operand value lengths	
Operand	Length
ACTION	1-8
COMMAND	1-90
DELAY	5-5
ENV	1-8
PARM	1-120

**Examples**

Two examples are provided to help you understand how to use the RUN command, and show the typical results.

**Example 1: Use the RUN command to execute an action, RSAMPLE, which sends a "Good afternoon" message to TS3163.**

```
GOMCMD OPMGRM1 DEFACTN NAME RSAMPLE COMMAND 'CP MSG TS3163 GOOD AFTERNOON!!!' ENV LVM
Ready; T=0.01/0.01 04:53:44

GOMCMD OPMGRM1 RUN ACTION RSAMPLE ENV LVM
04:54:03 * MSG FROM OPMGRM1 : GOOD AFTERNOON!!!
Ready; T=0.01/0.01 04:54:03
```

Figure 102. RUN command for RSAMPLE action

**Example 2: Use the RUN command to execute a Control Program (CP) command to send a "How are you today" message to TS3163 after 1 minute.**

```
GOMCMD OPMGRM1 RUN COMMAND 'CP MSG TS3163 HOW ARE YOU TODAY?' ENV LVM DELAY
01:00 Ready T=0.01/0.01 03:48:08
03:49:08 * MSG FROM OPMGRM1 : HOW ARE YOU TODAY?
```

Figure 103. Run command with CP message

## Usage notes

If the COMMAND operand is prefixed with the value 'CP ' (upper case CP, followed by a space), the command is processed through the CP diagnose x'0008' interface. The diagnose x'0008' interface may be used to send case-sensitive data to z/VM guests, such as Linux. The DEFOPTN value for MIXCASE must also be 'Y' to use case-sensitive data.

## Return codes

Table 92 on page 139 lists the return codes for the RUN command.

Table 92. RUN return codes	
Return code	Description
0	Finished correctly
-100	Error

## STATUS

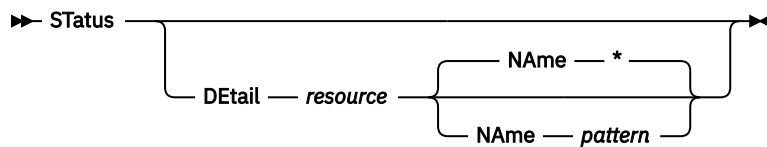


Figure 104. STATUS command syntax

## Authorization

The user issuing this command must be authorized for configuration or control commands.

## Purpose

Issue the STATUS command to show the current status of Operations Manager (GOMMAIN) processing.

- If the DETAIL operand is *not* present, a summary is provided as output.
- If the DETAIL operand is present, additional information about each specified resource is provided as output.

## Operands

### DEtail resource

Requests additional information for the specified resource definitions.

The valid resources are:

- ACTION — Displays the status of the defined actions.
- ACTIONQ — Displays the actions queued for execution.
- AUTH — Displays the status of authorized users.

- **CONSOLE** — Displays the status of the recording consoles.
- **DATASP** — Displays the status of the defined data space template sizes.
- **DATE** — Displays the status of the defined holidays.
- **EVENT** — Displays the status of the defined event monitors.
- **GROUP** — Displays the status of the defined groups.
- **IDLE** — Displays the status of the defined idle monitors.
- **IPCS** — Displays the status of the IP sessions.
- **IUCV** — Displays the status of the active IUCV connections.
- **MACHINE** — Displays the status of the monitored virtual machines.
- **OPTION** — Displays the status of the current options settings.
- **PAGE** — Displays the status of the defined page space monitors.
- **RULE** — Displays the status of the defined rules.
- **SCHEDULE** — Displays the status of the defined schedules.
- **SERVICE** — Displays the status of the defined action processing servers.
- **SPOOL** — Displays the status of the defined spool usage monitors.
- **SPOOLUSR** — Displays the top ten largest spool files and the top ten users with the most spool files.
- **VIEW** — Displays the status of the combined views.

#### **NAME pattern**

Lists information for resource names which match the pattern. Wildcard patterns are allowed. The wildcard pattern rules are explained in [Appendix B, “Wildcard pattern matching,” on page 181](#) .

#### **Related reference**

[DEFRULE](#)

## **Operand value lengths**

[Table 93 on page 140](#) lists the operand value lengths for the STATUS command.

<i>Table 93. STATUS operand value lengths</i>	
<b>Operand</b>	<b>Length</b>
DETAIL	1-8
NAME	1-8

## **Examples**

### **Example 1: Check the current status of Operations Manager (GOMMAIN) processing.**

The following examples show how to use the STATUS command. An explanation of the command and the results it generates are shown for each example.

```

GOMCMD OPMGRM1 STATUS
09/09/19 00:27:33 GOMCMD0899I OPERATIONS MANAGER V1R6 BUILT 2018/08/29 16:45
09/09/19 00:27:33 GOMCMD0900I 2 RULE(S) DEFINED
09/09/19 00:27:33 GOMCMD0902I 2 MACHINE MONITOR(S) DEFINED
09/09/19 00:27:33 GOMCMD0903I 4 SCHEDULE(S) DEFINED
09/09/19 00:27:33 GOMCMD0904I 5 ACTION(S) DEFINED
09/09/19 00:27:33 GOMCMD0905I 0 ACTION(S) SCHEDULED FOR PROCESSING
09/09/19 00:27:33 GOMCMD0906I 4 GROUP(S) DEFINED
09/09/19 00:27:33 GOMCMD0907I 0 DATES(S) DEFINED
09/09/19 00:27:33 GOMCMD0908I 0 SPOOL MONITOR(S) DEFINED
09/09/19 00:27:33 GOMCMD0909I 0 EVENT MONITOR(S) DEFINED
09/09/19 00:27:33 GOMCMD0910I 0 PAGE MONITOR(S) DEFINED
09/09/19 00:27:33 GOMCMD0911I 0 IDLE MONITOR(S) DEFINED
09/09/19 00:27:33 GOMCMD0920I 5 SERVICE MACHINE(S) DEFINED
09/09/19 00:27:33 GOMCMD0921I 4 USER(S)/GROUP(S) AUTHORIZED
09/09/19 00:27:33 GOMCMD0922I 4 IUCV CONNECTION(S)
09/09/19 00:27:33 GOMCMD0923I 0 DATA SPACE SIZE DEFINITION(S)
09/09/19 00:27:33 GOMCMD0924I 3 COMBINED VIEW(S)
09/09/19 00:27:33 GOMCMD0925I 13 USER CONSOLE(S) RECORDING
09/09/19 00:27:33 GOMCMD0926I 0 IP SESSION(S) DEFINED
Ready; T=0.01/0.01 00:27:33

```

Figure 105. Summary output using STATUS command without operands

**Example 2: Use STATUS with the DETAIL operand to list each user and the type(s) of authority currently assigned to the user(s).**

```

OMCMD OPMGRM1 STATUS DETAIL AUTH
09/09/19 00:32:06 GOMCMD2928I USER TS3163A AUTHORIZED FOR:
09/09/19 00:32:06 GOMCMD2995I CONTROL COMMANDS
09/09/19 00:32:06 GOMCMD2994I CONFIGURATION COMMANDS
09/09/19 00:32:06 GOMCMD2993I CONSOLE FOR *
09/09/19 00:32:06 GOMCMD2992I SPOOL FOR *
09/09/19 00:32:06 GOMCMD2928I USER TS3163 AUTHORIZED FOR:
09/09/19 00:32:06 GOMCMD2995I CONTROL COMMANDS
09/09/19 00:32:06 GOMCMD2994I CONFIGURATION COMMANDS
09/09/19 00:32:06 GOMCMD2993I CONSOLE FOR *
09/09/19 00:32:06 GOMCMD2992I SPOOL FOR *
09/09/19 00:32:06 GOMCMD2928I USER MAINT640 AUTHORIZED FOR:
09/09/19 00:32:06 GOMCMD2995I CONTROL COMMANDS
09/09/19 00:32:06 GOMCMD2994I CONFIGURATION COMMANDS
09/09/19 00:32:06 GOMCMD2993I CONSOLE FOR *
09/09/19 00:32:06 GOMCMD2992I SPOOL FOR *
09/09/19 00:32:06 GOMCMD2927I GROUP SYSADM AUTHORIZED FOR:
09/09/19 00:32:06 GOMCMD2995I CONTROL COMMANDS
09/09/19 00:32:06 GOMCMD2994I CONFIGURATION COMMANDS
09/09/19 00:32:06 GOMCMD2993I CONSOLE FOR *
09/09/19 00:32:06 GOMCMD2992I SPOOL FOR *
09/09/19 00:32:06 GOMCMD2929I INTERACTION FOR IP SESSION *
09/09/19 00:32:06 GOMCMD2999I STATUS DETAIL COMPLETE
Ready; T=0.01/0.01 00:33:09

```

Figure 106. STATUS command with DETAIL AUTH operand

**Example 3: Check the status of the recording consoles, such as the size and the number of messages.**

```
GOMCMD OPMGRM1 STATUS DETAIL CONSOLE
09/09/19 00:59:42 GOMCMD0965I CONSOLE SAMPLE2-ZVM6KT02 ASIT 00EDA5C0000000051 A
LET 0100000F SIZE 1M MSGS 1
09/09/19 00:59:42 GOMCMD0965I CONSOLE _GOMSCHD-ZVM6KT02 ASIT 00EDA9C0000000003 A
LET 0100000E SIZE 1M MSGS 33
09/09/19 00:59:42 GOMCMD0965I CONSOLE TMDMM-ZVM6KT02 ASIT 00EDA98000000000C A
LET 0100000D SIZE 1M MSGS 5
09/09/19 00:59:42 GOMCMD0965I CONSOLE TMLM1-ZVM6KT02 ASIT 00EDA8C0000000015 A
LET 0100000C SIZE 1M MSGS 5
09/09/19 00:59:42 GOMCMD0965I CONSOLE TMTMM-ZVM6KT02 ASIT 00EDA900000000069 A
LET 0100000B SIZE 1M MSGS 10
09/09/19 00:59:42 GOMCMD0965I CONSOLE KNAYAK-ZVM6KT02 ASIT 00EDA88000000005E A
LET 0100000A SIZE 1M MSGS 34
09/09/19 00:59:42 GOMCMD0965I CONSOLE SAMPLE-ZVM6KT02 ASIT 00EDA60000000005F A
LET 01000009 SIZE 1M MSGS 50
09/09/19 00:59:42 GOMCMD0965I CONSOLE _GOMMMON-ZVM6KT02 ASIT 00EDA70000000010E A
LET 01000008 SIZE 1M MSGS 7896
09/09/19 00:59:42 GOMCMD0965I CONSOLE OPERATOR-ZVM6KT02 ASIT 00EDAA00000000087 A
LET 01000007 SIZE 1M MSGS 1
09/09/19 00:59:42 GOMCMD0965I CONSOLE 5697J10F-ZVM6KT02 ASIT 00EDA5800000001DB A
LET 01000006 SIZE 1M MSGS 1
09/09/19 00:59:42 GOMCMD0965I CONSOLE TS3163-ZVM6KT02 ASIT 00EDA940000000073
LET 01000005 SIZE 1M MSGS 40
09/09/19 00:59:42 GOMCMD0965I CONSOLE MAINT640-ZVM6KT02 ASIT 00EDA8000000000CC A
LET 01000004 SIZE 1M MSGS 39
09/09/19 00:59:42 GOMCMD0965I CONSOLE RACFVM-ZVM6KT02 ASIT 00EDA7C000000000D8 A
LET 01000003 SIZE 1M MSGS 4
09/09/19 00:59:42 GOMCMD2999I STATUS DETAIL COMPLETE
Ready; T=0.01/0.01 01:00:02
```

Figure 107. STATUS command with DETAIL CONSOLE operands

**Example 4: Check the status and the details of monitored virtual machines.**

```
STATUS DETAIL (MACHINE)
2019-05-26 11:46:16 GOMCMD0942I MACHINE MONITOR MON1 TRIGGERED 0 TIME(S)
2019-05-26 11:46:16 GOMCMD2977I ACTION MACHAC
2019-05-26 11:46:16 GOMCMD2976I DAYS ALL
2019-05-26 11:46:16 GOMCMD2975I HOURS ALL
2019-05-26 11:46:16 GOMCMD2973I USER RSCS (RSCS - DSC)
2019-05-26 11:46:16 GOMCMD2972I INTERVAL 1 MINUTE(S)
2019-05-26 11:46:16 GOMCMD2971I DELAY 0 MINUTE(S)
2019-05-26 11:46:16 GOMCMD2970I RETRY INTERVAL 1 MINUTE(S) RETRY COUNT 0
2019-05-26 11:46:16 GOMCMD2968I NEXT CHECK 2019/05/26 11:46
2019-05-26 11:46:16 GOMCMD2999I STATUS DETAIL COMPLETE
```

Figure 108. Details about monitored virtual machines

**Example 5: Check the status and details of spool usage monitors.**

```
STATUS DETAIL (SPOOL)
2019-05-26 11:48:49 GOMCMD0948I SPOOL MONITOR WARN TRIGGERED 0 TIME(S)
2019-05-26 11:48:49 GOMCMD2977I ACTION ACTNSP
2019-05-26 11:48:49 GOMCMD2976I DAYS ALL
2019-05-26 11:48:49 GOMCMD2975I HOURS ALL
2019-05-26 11:48:49 GOMCMD2967I USAGE LOW 80 HIGH 89
2019-05-26 11:48:49 GOMCMD2972I INTERVAL 1 MINUTE(S)
2019-05-26 11:48:49 GOMCMD2968I NEXT CHECK 2019/05/26 11:48
2019-05-26 11:48:49 GOMCMD0948I SPOOL MONITOR CRIT TRIGGERED 0 TIME(S)
2019-05-26 11:48:49 GOMCMD2977I ACTION ACTNSP
2019-05-26 11:48:49 GOMCMD2976I DAYS ALL
2019-05-26 11:48:49 GOMCMD2975I HOURS ALL
2019-05-26 11:48:49 GOMCMD2967I USAGE LOW 90 HIGH 100
2019-05-26 11:48:49 GOMCMD2972I INTERVAL 1 MINUTE(S)
2019-05-26 11:48:49 GOMCMD2968I NEXT CHECK 2019/05/26 11:48
2019-05-26 11:48:49 GOMCMD2999I STATUS DETAIL COMPLETE
```

Figure 109. Details about all spool usage monitors

### Example 6: Check the status and details of users' spool files.

```
status detail spoolusr
2019-05-26 11:51:11 GOMCMD0970I USER OPERATNS SPOOL FILE ID 0175 IS USING 8401 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0970I USER RWHITE2 SPOOL FILE ID 0445 IS USING 7192 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0970I USER RWHITE2 SPOOL FILE ID 0446 IS USING 7192 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0970I USER SAMARAK SPOOL FILE ID 0070 IS USING 6167 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0970I USER RWHITE2 SPOOL FILE ID 0448 IS USING 6030 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0970I USER SAMARAK SPOOL FILE ID 0021 IS USING 5051 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0970I USER SMARAK SPOOL FILE ID 0003 IS USING 4226 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0970I USER RWHITE2 SPOOL FILE ID 0450 IS USING 3483 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0970I USER PERFSVM SPOOL FILE ID 0383 IS USING 2986 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0970I USER PERFSVM SPOOL FILE ID 0380 IS USING 2985 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0971I USER BKRADMIN HAS 1673 SPOOL FILES USING 59605 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0971I USER DMARTIN HAS 609 SPOOL FILES USING 23093 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0971I USER MAINT HAS 351 SPOOL FILES USING 1713 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0971I USER SAMARAK HAS 229 SPOOL FILES USING 88504 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0971I USER TCPMAINT HAS 218 SPOOL FILES USING 2873 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0971I USER MAINT640 HAS 217 SPOOL FILES USING 994 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0971I USER CSHOWA HAS 215 SPOOL FILES USING 879 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0971I USER CSHOW2 HAS 207 SPOOL FILES USING 1262 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0971I USER OPERATOR HAS 207 SPOOL FILES USING 10618 SPOOL BLOCKS
2019-05-26 11:51:11 GOMCMD0971I USER MDMARA HAS 199 SPOOL FILES USING 4599 SPOOL BLOCKS
```

Figure 110. Details about users' spool files

## Usage notes

None

## Return codes

Table 94 on page 143 lists the return codes for the STATUS command.

Table 94. STATUS return codes	
Return code	Description
0	Finished correctly
-100	Error

## SUSPEND

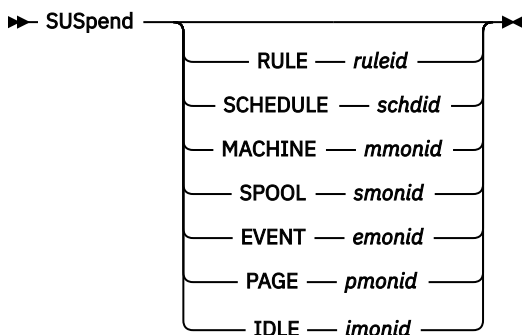


Figure 111. SUSPEND command syntax

## Authorization

The user issuing this command must be authorized for configuration commands.

## Purpose

Issue the SUSPEND command to deactivate or suspend a resource, such as a rule, schedule, or monitor. A resource in the suspended state is ignored for processing.

The valid resources are:

- RULE
- MACHINE
- SCHEDULE
- SPOOL
- EVENT
- PAGE
- IDLE

## Operands

### **RULE**

Specifies the *ruleid* from DEFRULE.

### **MACHINE**

Specifies the *mmonid* from DEFMMON.

### **SCHEDULE**

Specifies the *schdid* from DEFSCHD.

### **SPOOL**

Specifies the *smonid* from DEFSMON.

### **EVENT**

Specifies the *emonid* from DEFEMON.

### **PAGE**

Specifies the *pmonid* from DEFPMON.

### **IDLE**

Specifies the *imonid* from DEFIMON.

### **Related reference**

[DEFEMON](#)

[DEFIMON](#)

[DEFMMON](#)

[DEFPMON](#)

[DEFRULE](#)

[DEFSCHD](#)

[DEFSMON](#)

[RESUME](#)

## Operand value lengths

Table 95 on page 144 lists the operand value lengths for the SUSPEND command.

Table 95. SUSPEND operand value lengths	
Operand	Length
EVENT	1-8
IDLE	1-8
MACHINE	1-8



Table 95. SUSPEND operand value lengths (continued)	
Operand	Length
PAGE	1-8
RULE	1-8
SCHEDULE	1-8
SPOOL	1-8

## Example

An example is provided to help you understand how to use the SUSPEND command, and shows the typical results of this command.

### Example 1: Use the SUSPEND command to deactivate or suspend a spool usage monitor, SMON, so that it is ignored for processing.

In the example below, notice that the SUSPEND command is at the bottom of this example. Due to low usage, the SMON spool monitor is being suspended.

```
GOMCMD OPMGRM1 DEFACTN NAME ASPOOL COMMAND 'CP MSG TS3163 SAYS SPOOL IS &4% FULL',ENV
LVM
Ready; T=0.01/0.01 04:12:48

GOMCMD OPMGRM1 DEFSMON NAME SMON ACTION ASPOOL USAGE 000-002
Ready; T=0.01/0.01 04:12:57

04:13:00 * MSG FROM OPMGRM1 : SAYS SPOOL IS 1% FULL
04:13:30 * MSG FROM OPMGRM1 : SAYS SPOOL IS 1% FULL
04:14:30 * MSG FROM OPMGRM1 : SAYS SPOOL IS 1% FULL
04:15:30 * MSG FROM OPMGRM1 : SAYS SPOOL IS 1% FULL

GOMCMD OPMGRM1 SUSPEND SPOOL SMON
Ready; T=0.01/0.01 04:15:35
```

Figure 112. SUSPEND command issued

## Usage notes

1. One of the resources must be specified.
2. Only one of the resources must be specified.
3. The use of wildcard characters is not allowed.
4. Use ALL in place of ruleid, mmonid, schdid, smonid, emonid, pmonid, or imonid to suspend all definitions for the specified resource type. For example, the following command suspends all defined rules:

```
SUSPEND RULE ALL
```

## Return codes

Table 96 on page 145 lists the return codes for the RESUME command.

Table 96. SUSPEND return codes	
Return Code	Description
0	Finished correctly
-100	Error



---

## Chapter 9. Syntax for console, log, and spool access commands and commands for IP session interaction

This chapter describes the Operations Manager console, log, and spool access commands and their syntax. It also describes commands that apply to using IP sessions.

---

### Purpose of console, log, and spool access commands

Console, log, and spool access commands allow you to view and modify a user's console display, the Operations Manager log, and a user's spool files. With a console, log, or spool access command, you can perform various tasks such as issue commands on a user's console, display the console or log in full-screen mode, modify the attributes in a console, add comments in a log, and view a spool list. You can also create a combined view.

Use GOMCMD to issue console, log, and spool access commands.

#### Related reference

[GOMCMD](#)

---

### Authorization for console, log, and spool access commands

Authorization varies for console, log and spool access commands. Refer to the specific command for the required authorization.

---

### Console data spaces

Operations Manager uses z/VM data spaces to contain several types of output.

The data spaces may contain any of the following:

- Console output from a monitored user ID (for example, output sent to Operations Manager via SECUSER or OBSERVER).
- Combined views of console output defined in the DEFVIEW command.
- The Operations Manager log.
- Output and information related to the execution of schedules, machine monitors, spool monitors, event monitors, and the output from actions.

Use the VIEWCON command to look at any of these data spaces except the logging data space. You must use the VIEWLOG command to look at the Operations Manager log data space.

Table 97 on page 147 lists reserved data space names and the corresponding execution type.

Table 97. Reserved data space names	
Execution type	Data space name
Event monitors	_GOMEMON
Idle monitors	_GOMIMON
Log	_GOMALOG
Machine monitors	_GOMMMON
Output from actions	_GOMACTN
Page space monitors	_GOMPMON

Table 97. Reserved data space names (continued)	
Execution type	Data space name
Schedules	_GOMSCHD
Spool monitors	_GOMSMON

#### Related reference

[VIEWCON](#)

[VIEWLOG](#)

## ALTRCON

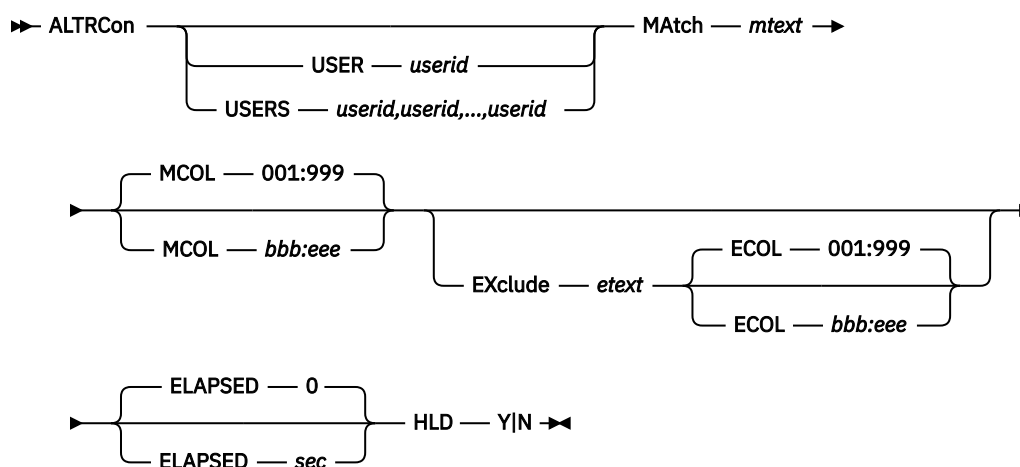


Figure 113. ALTRCON command syntax

## Authority

The user issuing this command must be authorized for console access for the specified userid or view.

## Purpose

Use the ALTRCON command to modify the attributes of lines in a user's console when it is displayed using the VIEWCON command.

The ALTRCON command can also be issued to modify the attributes of lines in a combined view created with a DEFVIEW command.

#### Related reference

[DEFVIEW](#)

[DEFRULE](#)

[VIEWCON](#)

## Operands

#### **USER *userid***

Specifies the *userid* or view name for the console you want to modify.

#### **USERS *userid,userid,...,userid***

Specifies up to 16 *userids* (consoles) to modify. The *userid* can be a wildcard pattern. For more information on the wildcard rules, refer to [Appendix B, "Wildcard pattern matching,"](#) on page 181.

**MATCH *mtext***

Specifies the pattern used to compare against the lines to determine if there is a match. Wildcard pattern matching is allowed. For more information on the wildcard rules, refer to [Appendix B, “Wildcard pattern matching,”](#) on page 181.

**MCOL *bbb:eee***

Specifies the beginning and ending columns to search for the match pattern (inclusive). The format is *bbb:eee*, where *bbb* is the beginning column and *eee* is the ending column.

**EXCLUDE *etext***

Specifies the pattern that will be used to compare against the lines to determine if there is an exclusion. Wildcard pattern matching is allowed. For more information on the wildcard rules, refer to [Appendix B, “Wildcard pattern matching,”](#) on page 181.

**ECOL *bbb:eee***

Specifies the beginning and ending columns to search for the exclude pattern (inclusive). The format is *bbb:eee*, where *bbb* is the beginning column and *eee* is the ending column.

**ELAPSED *sec***

Specifies the minimum amount of elapsed time since a line was logged for that line to be eligible for modification. Only lines which have been in the console longer than the number of elapsed seconds are eligible for modification. The value must be between 0 and 999999.

**HLD *Y/N***

Specifies the setting of the HLD attribute.

## Operand value lengths

Table 98 on page 149 lists the operand value lengths for the ALTRCON command.

<i>Table 98. ALTRCON operand value lengths</i>	
<b>Operand</b>	<b>Length</b>
ECOL	7
ELAPSED	1-6
EXCLUDE	1-60
HLD	1
MATCH	1-60
MCOL	7
USER	1-8
USERS	1-120

## Examples

An example is provided to help you understand how to use the ALTRCON command to alter or modify attributes of lines displayed on a user's console.

**Example 1: When the console of user MAINT640 is displayed using the VIEWCON command, issue the ALTRCON command to find "CLEAR" and hold that line.**

Issue the following command to find a match for the text, CLEAR (or clear), in columns 55 through 61 in the text displayed on the user console. If CLEAR is found, hold that line to prevent it from autoscrolling.

```
GOMCMD OPMGRM1 ALTRCON USER MAINT640 MATCH CLEAR MCOL 055:61 HLD Y
```

On the screen below, the console of user MAINT640 is shown and the line containing the word "clear" is held.

```

logoff hold
CONNECT= 00:23:42 VIRTCPU= 000:00.00 TOTCPU= 000:00.01
LOGOFF AT 04:08:51 CDT WEDNESDAY 09/04/19
Press enter or clear key to continue
RDR FILE 0454 SENT FROM MAINT640 CON WAS 0454 RECS 0081 CPY 001 T NOHOLD NOKEE

```

Figure 114. ALTRCON command issued

## Usage notes

Either USER or USERS must be specified.

## Return codes

Table 99 on page 150 lists the return codes for the ALTRCON command.

Table 99. ALTRCON return codes	
Return code	Description
0	Finished correctly
-100	Error

## CMDIPCS

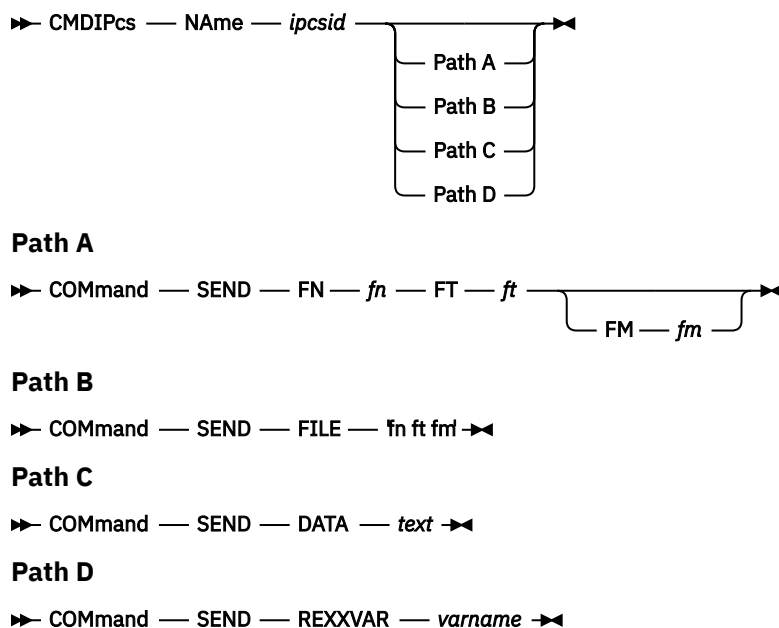


Figure 115. CMDIPCS command syntax

## Authority

The user issuing this command must be authorized for interaction with the IP session specified by the *NAME* operand.

## Purpose

Use the CMDIPCS command to send a command request to an IP communication session which is processing a program.

## Operands

### **NAME**

Specifies the name of the IP session. The value must match the NAME operand's value that is specified in a DEFIPCS command.

### **Path A, B, C, D**

This path sends requests to an IP session defined with DEFIPCS and a value of GOMRSGW for the APPL keyword.

### **COMMAND SEND**

Sends data from a file, operand, or REXX variable.

### **FN *fn***

Specifies the filename of the file to send.

### **FT *ft***

Specifies the filetype of the file to send.

### **FM *fm***

Specifies the filemode of the file to send. The default is any accessed CMS disk.

### **FILE *fn ft fm***

Specifies the filename, filetype, and filemode of the file to send. Values for *fn* and *ft* are required. If *fm* is omitted, the default is any accessed CMS disk.

### **DATA *text***

Specifies the text to send. If the text consists of only one word, the value may be entered without any single quotes. For example:

```
DATA jump
```

However, if the text contains two or more words, single quotes are required:

```
DATA 'jump ship'
```

### **REXXVAR *varname***

Specifies a REXX variable name containing the data to send.

## Example

Refer to the EXAMPLES CONFSAMP file and the SENDSYSL SAMPEXEC file on the 5697J10F 2C2 (samples) disk for examples on how to use the CMDIPCS command.

## Usage notes

1. The GOMRSGW application supports the SEND command.
2. Paths A, B, C, and D are mutually exclusive, meaning operands from two or more paths may not be combined in the same command.
3. For both Path A and Path B, the file specified must be accessible by the machine running GOMMAIN, typically OPMGRM1.
4. Path D is only allowed when the CMDIPCS command is issued within a REXX EXEC on the machine running GOMMAIN, typically OPMGRM1. To ensure the action is performed only on the machine running GOMMAIN, specify ENV LVM on the DEFACTN command.
5. A sample file, sendsys1.sample, can be found in the SAMPLIB. Use this sample file to help you create customized statements or EXEC files for your site and for sending system logs. Refer to [Appendix A, "Sample files," on page 177](#) for more information.

## LOGTEXT

➤ LOGTEXT — TEXT *comment* ➤

Figure 116. LOGTEXT command syntax

## Authorization

No specific authority is required for this command.

## Purpose

Issue the LOGTEXT command to add a comment to the Operations Manager log file.

### Related reference

[VIEWLOG](#)

## Operands

### TEXT *comment*

Specifies the comment text that appears in the log. The value may be entered as 'comment'.

## Operand value lengths

Table 100 on page 152 lists the operand value lengths for the LOGTEXT command.

Table 100. LOGTEXT operand value lengths	
Operand	Length
TEXT	1-120

## Example

An example is provided below to help you see how to use the LOGTEXT command and the typical results.

### Example 1: Add the 'text' listed in the LOGTEXT operand to the Operations Manager log.

For example, see the following command example:

```
GOMCMD OPMGRM1 LOGTEXT TEXT 'text'
```

In place of 'text', you can provide other text, such as:

```
'ADDING DETAILS'
```

To see the result, you can view the text you added to the log by specifying the following command:

```
GOMCMD OPMGRM1 VIEWLOG
```

An example of the resulting message is shown below.

```
09/05/19 01:19:52 GOMCMD0204I  ADDING DETAILS
```

## Usage notes

None

## Return codes

Table 101 on page 153 lists the return codes for the LOGTEXT command.



Table 101. LOGTEXT return codes	
Return code	Description
0	Finished correctly
-100	Error

## VIEWCON

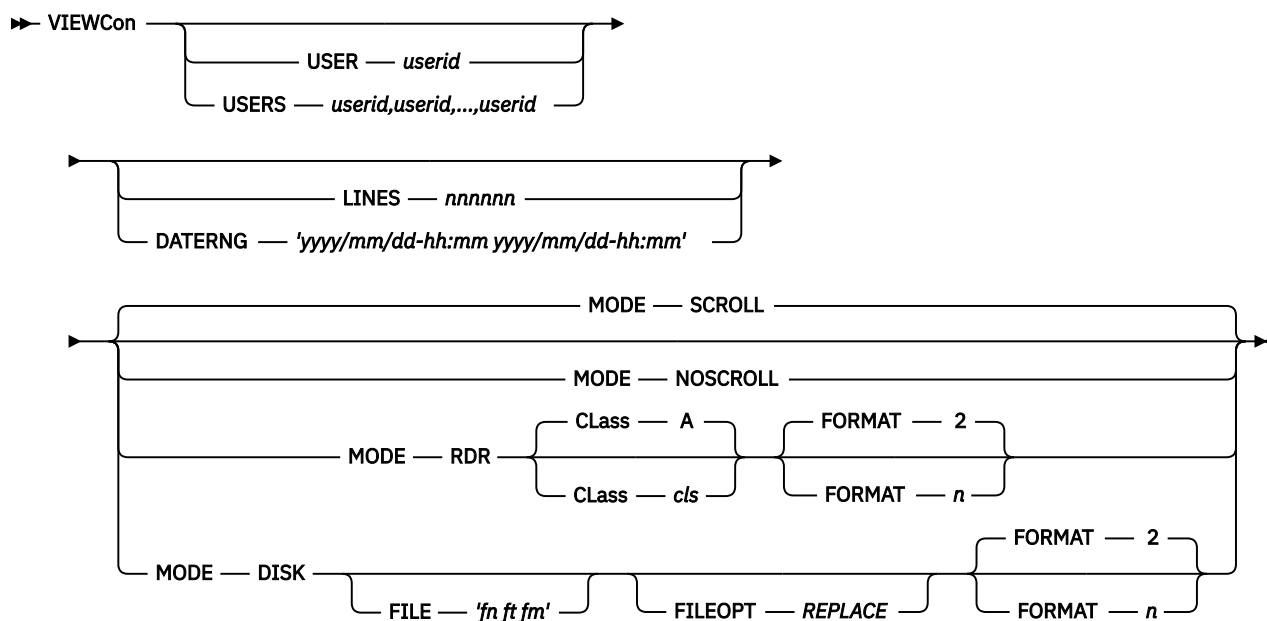


Figure 117. VIEWCON command syntax

## Authorization

The user issuing this command must be authorized for console access for the specified `userid(s)` or view.

### Related tasks

[Step 5. \(Optional\) Configure an external security mechanism](#)

## Purpose

Use the VIEWCON command to display the current console content for a monitored user in full-screen mode and to issue commands on that user's console or place a copy of the current console in your virtual reader.

The VIEWCON command may also be issued to display a combined view created with a DEFVIEW command. Commands may be issued to any monitored user combined into the view. The target `userid` and a space must precede the command.

## Operands

### USER *userid*

Specifies the user ID or view name for the console you want to display.

There are five reserved console data spaces available to use with VIEWCON USER. Refer to [“Console data spaces”](#) on page 147 for more information on the reserved console data space names.

**Note:** You cannot use VIEWCON USER to view the logging data space. You must use the VIEWLOG command to view the logging console.

**USERS *userid,userid,...,userid***

Specifies the *userid*s for up to sixteen (16) consoles to view. The *userid* can be a wildcard pattern. The *userid* is not matched against views. The console information is presented by creating a combined view for the *userid*s. The wildcard rules are explained in [Appendix B, “Wildcard pattern matching,”](#) on page 181.

**LINES *nnnnnn***

Specifies the number of lines you wish to view. The last *nnnnnn* lines are displayed. The default is to view all lines.

**DATERNG '*yyyy/mm/dd-hh:mm yyyy/mm/dd-hh:mm*'**

Specifies the beginning and ending date and time range of lines to display within the current console content. The default is to view all lines of the current console content. If the ending date and time specified is earlier than the current date and time, no new messages will be displayed.

**MODE SCROLL**

Specifies that the initial view is autoscrolling. SCROLL is the default.

**MODE NOSCROLL**

Specifies that the initial view is *not* autoscrolling.

**MODE RDR**

Specifies that a copy of the current console is to be placed in your virtual reader.

**MODE DISK**

Specifies that a copy of the current console is to be placed in a CMS file *systemname userid A1*. Use the **FILE** option to specify a different filename, filetype, and filemode.

**Class *cls***

Specifies the reader file class. The default is class A.

**FORMAT *n***

Specifies the value for line header format. The values are listed under the FORMAT field of VIEWCON PROFILE.

**FILE '*fn ft fm*'**

Specifies the filename, filetype, and filemode to place the current console.

The value for *fn* is required.

If *ft* is omitted, the default is YYYYMMDD.

If *fm* is omitted, the default is A1.

**FILEOPT REPLACE**

Specifies that the output file to replace an existing file with the same file identifier.

If the file already exists, the command will end with an error message, unless the FILEOPT REPLACE option is specified. The operation will terminate with the Return code RC-4.

**Related concepts**

[Wildcard pattern matching](#)

Wildcard pattern matching is used by several of the commands. Wildcards allow you to search for and process characters that are similar.

**Related reference**

[AUTH](#)

[DEFDSPS](#)

[DEFOPTN](#)

[DEFVIEW](#)

## Operand value lengths

[Table 102 on page 155](#) lists the operand value lengths for the VIEWCON command.

Table 102. VIEWCON operand value lengths	
Operand	Length
CLASS	1
LINES	1-6
MODE	3-8
USER	1-8
USERS	1-120

## Examples

This section provides examples on how to use the VIEWCON command.

### Example 1: Use VIEWCON to view the current console of a specific monitored user, TS3163.

```
GOMCMD OPMGRM1 VIEWCON USER TS3163

HCPMID6001I  TIME IS 00:00:00 CDT SATURDAY 08/31/19
HCPMID6001I  TIME IS 00:00:00 CDT SUNDAY 09/01/19
HCPMID6001I  TIME IS 00:00:00 CDT MONDAY 09/02/19

TS3163-ZVM6KT02  (Scroll)
```

Figure 118. VIEWCON command

### Example 2: View the current console of a monitored user, MAINT640, and issue commands on that user's console.

```
GOMCMD OPMGRM1 VIEWCON USER MAINT640

HCPMID6001I  TIME IS 00:00:00 CDT SATURDAY 08/31/19
HCPMID6001I  TIME IS 00:00:00 CDT SUNDAY 09/01/19
HCPMID6001I  TIME IS 00:00:00 CDT MONDAY 09/02/19
RDR FILE 0442 SENT FROM MAINT640 CON WAS 0442 RECS 0060 CPY 001 T NOHOLD NOKEEP

MAINT640-ZVM6KT02  (Scroll)
```

Figure 119. VIEWCON command

You can now issue a command, such as Q DISK, to query the contents on a disk associated with this user.

### Example 3: Use the VIEWCON command to see a combined view that is first created with a DEFVIEW command.

Before issuing the VIEWCON command, you must first use the DEFVIEW command. In this example, the DEFVIEW command defines a combined view, SAMPLE, that includes multiple user ID's, which are MAINT640 and TS3163.

```
GOMCMD OPMGRM1 DEFVIEW NAME SAMPLE USER MAINT640
Ready; T=0.01/0.01 06:05:00
GOMCMD OPMGRM1 DEFVIEW NAME SAMPLE USER TS3163
Ready; T=0.01/0.01 06:05:10
```

Figure 120. Example of DEFVIEW for multiple user ID's

Next, issue the VIEWCON command for the combined view, SAMPLE.

```
GOMCMD OPMGRM1 VIEWCON USER SAMPLE
```

Figure 121. Example of VIEWCON for combined view

User console content from both MAINT640 and TS3163 are displayed together in a single screen.

## Usage notes

1. Either USER or USERS must be specified.
2. The date format displayed in VIEWCON will be based on the CP DATEFORMAT setting of the user issuing the VIEWCON command. If MODE(RDR) is specified, the date format will be provided in yyyy/mm/dd format.
3. LINES and DATERNG are mutually exclusive.
4. The name of the combined view created when using the USERS operand begins with \$TMP. See the “DEFDSPS” on page 50 command to control the data space size for the combined view.
5. For an explanation of wildcard pattern matching, see [Appendix B, “Wildcard pattern matching,” on page 181](#).
6. You can use the following default PF keys to move through the console display:
  - PF01 — Show the help screen.
  - PF02 — Pass the current VIEWCON session contents to XEDIT.
  - PF03 — End the VIEWCON session.
  - PF05 — Toggle the acknowledgment for a line in the console. To clear an existing hold status, press the PF05 key again.
  - PF06 — Toggle the date/time format displayed.
  - PF07 — Scroll up.
  - PF08 — Scroll down.
  - PF10 — Scroll left.
  - PF11 — Scroll right.
  - PF12 — Retrieve previous command.
  - PF19 — Scroll to the top.
  - PF20 — Scroll to the bottom.
  - PF21 — Process CMS command.
  - PF22 — Display the users viewing the console.
  - PF23 — Toggle auto scrolling.
  - PF24 — Display the PF key assignments.
  - PA02 — Toggle auto scrolling.
  - ENTER — Process the command on the command line.

**Note:** For the scrolling PF keys, you can enter a scrolling amount on the command line. If no amount is specified, the scroll amount is equal to the size of the viewing area.

A command may be typed on the command line for the process CMS command action.

7. To invoke XEDIT to review the current content, you must have sufficient disk space to **save** the current content. The filename and filetype of the file created by Operations Manager will be *userid* *systemname*. If a file already exists, it will be overwritten.

The default filemode of the file created by Operations Manager is **A1**. The XEDIT rules for filemode processing remain in effect. Thus, using the default filemode A1 will result in the file remaining on the A-disk upon exit.

You may control the file mode used to save the current content by using the XEDITFM=*fm* setting in VIEWCON PROFILE. For example, by using XEDITFM=A3, if you do not save any changes to the file during the XEDIT session, it will be erased upon exit.

Any changes made while using XEDIT are not reflected in the live view upon returning from XEDIT.

8. If an external security manager is being used, for example `DEFOPTN ESM(Y) , AUTHCMD(N)` or `DEFOPTN ESM(Y) , AUTHCMD(Y)`, the user must have READ access to the profile `ESMPREF.ALTRCON.userid` for authorization to use the HOLD/UNHOLD toggle function. Refer to [“ESM class and Facility class profiles” on page 11](#) for more information. If an external security manager is not being used, no authorization checking is done for the HOLD/UNHOLD toggle function.
9. The ALTRCON command and the VIEWCON HOLD/UNHOLD toggle perform the same function.
10. The amount of data displayed or sent to your virtual reader is set by the user's console or view data space size.
11. To search through the display, use the forward slash (/) to search forward and the back slash (\) to search backward. Use the equal sign (=) to repeat the previous command. Issue the DEFOPTN command to change these characters. See [“DEFOPTN” on page 57](#) for more information.
12. The actions for PF01–PF24, initial value for the date/time format display, and display of the PF key information lines are configurable through the VIEWCON PROFILE file.
  - a. Issue the following command in the VIEWCON PROFILE file to customize the actions for PF01–PF24:

```
PFnn=command
```

Where *nn* is 01 - 24 and *command* is a valid PF key action or command.

The valid PF key actions are:

- HELP — Show the help screen.
- XEDIT — Pass the current VIEWCON session contents to XEDIT.
- END — End the VIEWCON session.
- HOLD — Toggle the acknowledgment for a line in the console.
- FORMAT — Toggle the date/time format displayed.
- UP — Scroll up.
- DOWN — Scroll down.
- LEFT — Scroll left.
- RIGHT — Scroll right.
- RECALL — Retrieve previous command.
- TOP — Scroll to top.
- BOT — Scroll to bottom.
- EXCMD — Process CMS command (the command to process may be included).

Refer to Usage Note [“13” on page 158](#) for more information.

- USERS — Display the users viewing the console.
- SCROLL — Toggle auto scrolling.
- VIEWPF — Display the PF key assignments.

- b. Issue one of the following commands in the VIEWCON PROFILE file to customize the initial date/time format display:
  - FORMAT=0 — Displays the console data with no date and no time. This is the default.
  - FORMAT=1 — Displays the console data with time.
  - FORMAT=2 — Displays the console data with date and time.
  - FORMAT=3 — Displays the console data with date, time and message class.
  - FORMAT=4 — Displays the console data with time and message class.
  - FORMAT=5 — Displays the console data with message class.

- c. Issue the following command in the VIEWCON PROFILE file to control the display of the PF key information lines:

```
SHOWPF=c[,h,xat,clr]
```

Where *c* is one of the following options:

- N — Do not display the PF key assignments on the screen.
- Y — Display PF01–PF12 assignments on the screen.
- A — Display PF01–PF24 assignments on the screen.

The values for *h,xat,clr* are optional. If one of the values is specified, they must all be specified.

- *h* — Specifies if the line is highlighted. Values are Y or N.
- *xat* — Specifies the extended attributes applied to the line. Values are:
  - AAI — Leave as is
  - ABL — Blinking
  - ARV — Reverse video
  - AUL — Underline
- *clr* — Specifies the color applied to the line. Values are:
  - CAI — Leave as is
  - CBL — Blue
  - CRE — Red
  - CPI — Pink
  - CGR — Green
  - CCY — Cyan
  - CYE — Yellow
  - CWH — White

- d. Issue the following command in the VIEWCON PROFILE file to control the location, sharing, and disposition of the contents passed to XEDIT:

```
XEDITFM=m[n]
```

where *m* is the file mode *letter* and *n* is the file mode *number*.

- The minidisk/SFS accessed as *m*, must be R/W (read/write).
- The file mode number may be any valid CMS file number. You can specify a file mode number of 3 and the file passed to XEDIT will be erased by XEDIT after reading the file. If you specify a file mode number of 3, you must use the SAVE/FILE XEDIT command to retain the file.

The default is XEDITFM=A1.

The following shows an example of what might be in a VIEWCON PROFILE file after customizing a PF key assignment, the initial date/time format, and the display of the PF keys:

```
PF13=HELP  
FORMAT=1  
SHOWPF=Y,Y,AAI,CAI
```

13. While using VIEWCON, you can also execute a z/VM or Operations Manager command by using EXCMD. Execute the EXCMD command with one of the following methods:

- Press a function key that is assigned to EXCMD without any command on the command line. This invokes CMS SUBSET. To return to VIEWCON, use the CMS command return.
- Press a function key that is assigned to EXCMD with a CMS command on the command line.

When the command completes, if the command on the command line is not an Operations Manager command, you return to the originating VIEWCON command.

When the command completes, if the command on the command line is an Operations Manager command, you return to the point that originated the VIEWCON command.

- Assign EXCMD with a z/VM or Operations Manager command to a PF key in VIEWCON PROFILE. For example, PF19=EXCMD LISTFILE \* \* A or PF20=EXCMD GOMCMD OPMGRM1 VIEWLOG.

When the command completes, if the command assigned to the PF key is not an Operations Manager command, you return to the originating VIEWCON command.

When the command completes, if the command assigned to the PF key is an Operations Manager command, you return to the point that originated the VIEWCON command.

Return codes

Table 103 on page 159 lists the return codes for the VIEWCON command.

Table 103. VIEWCON return codes	
Return code	Description
0	Finished correctly
-100	Error

VIEWLOG

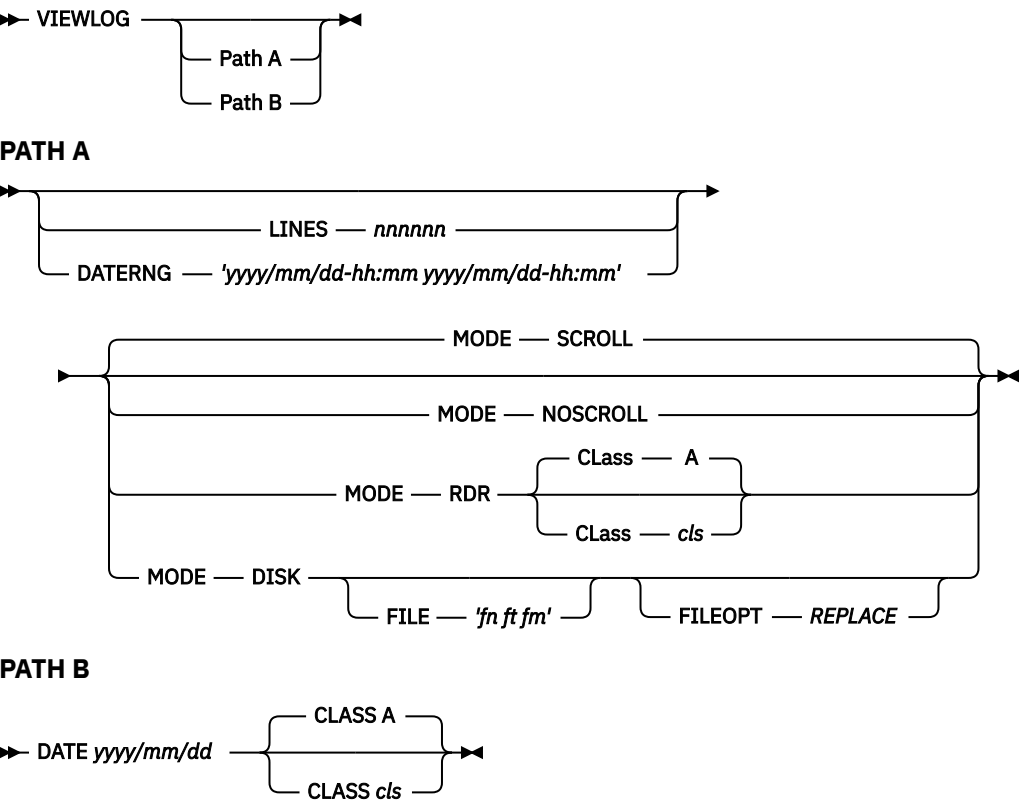


Figure 122. VIEWLOG command syntax

Authorization

The user issuing this command must be authorized for control commands.

## Purpose

Use the VIEWLOG command to display the current Operations Manager log in full-screen mode and to issue Operations Manager commands or place a copy of the log in your virtual reader.

You can also issue the VIEWLOG command to obtain a copy (a DISK DUMP) of the log for a specific day.

## Operands

### **LINES** *nnnnnn*

Specifies the number of lines you wish to view. The last *nnnnnn* lines are displayed. The default is to view all lines.

### **DATERNG** '*yyyy/mm/dd-hh:mm yyyy/mm/dd-hh:mm*'

Specifies the beginning and ending date and time range of lines to display within the current console content. The default is to view all lines of the current log content. If the ending date and time are earlier than the current date and time, no new lines will be displayed.

### **MODE** **SCROLL**

Specifies the initial view is autoscrolling. SCROLL is the default.

### **MODE** **NOScroll**

Specifies the initial view is *not* autoscrolling.

### **MODE** **RDR**

Specifies that a copy of the active Operations Manager log (VIEWLOG ) is to be placed in your virtual reader.

### **MODE** **DISK**

Specifies that a copy of the current Operations Manager log is to be placed in a CMS file *systemname* *userid* A1. Use the **FILE** option to specify a different filename, filetype, and filemode.

### **DATE** *yyyy/mm/dd*

Specifies that a copy of the log for an entire day is to be placed in your virtual reader. Use the format *yyyy/mm/dd*, where:

- *yyyy*. The year, which must be specified with four digits.
- *mm*. The month (01-12).
- *dd*. The day of the month (01-31).

### **Class** *cls*

Specifies the reader file class. Class A is the default.

### **FILE** '*fn ft fm*'

Specifies the filename, filetype, and filemode to place the current Operations Manager log.

The value for *fn* is required.

If *ft* is omitted, the default is YYYYMMDD.

If *fm* is omitted, the default is A1.

### **FILEOPT** **REPLACE**

Specifies that the output file to replace an existing file with the same file identifier.

If the file already exists, the command will end with an error message, unless the FILEOPT REPLACE option is specified. The operation will terminate with the Return code RC-4.

## Operand value lengths

Table 104 on page 160 lists the operand value lengths for the VIEWLOG command.

Table 104. VIEWLOG operand value lengths	
Operand	Length
CLASS	1



Table 104. VIEWLOG operand value lengths (continued)	
Operand	Length
DATE	10
LINES	1-6
MODE	3-8

## Examples

### Example 1: Use the VIEWLOG command to display the current Operations Manager log.

```
GOMCMD OPMGRM1 VIEWLOG
09/09/19 01:12:09 GOMIUC0188L IUCV SYSTEM AND USER SET SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003
USR=TS3163 0003
09/09/19 01:12:09 GOMCMD0201L "VIEWSPL DATE 2019/09/05" SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003
USR=TS3163 SRC=MASIUCV CLS=99
09/09/19 01:15:35 GOMIUC0186L IUCV DISCONNECTING SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003 USR=TS3163
0003
09/09/19 01:15:36 GOMIUC0185L IUCV ACCEPTING SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003 USR=TS3163
0003
09/09/19 01:15:36 GOMIUC0188L IUCV SYSTEM AND USER SET SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003
USR=TS3163 0003
09/09/19 01:15:36 GOMCMD0201L "VIEWSPL SIZE 144K" SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003
USR=TS3163 SRC=MASIUCV CLS=99
09/09/19 01:16:49 GOMIUC0186L IUCV DISCONNECTING SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003 USR=TS3163
0003
09/09/19 01:19:32 GOMIUC0185L IUCV ACCEPTING SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003 USR=TS3163 0003
09/09/19 01:19:32 GOMIUC0188L IUCV SYSTEM AND USER SET SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003
USR=TS3163 SRC=MASIUCV CLS=99
09/09/19 01:19:32 GOMCMD0201L "VIEWSPL SFID MAINT640,0068" SYS=ZVMTESTS VID=TS3163
09/09/19 01:22:27 GOMIUC0186L IUCV DISCONNECTING SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003 USR=TS3163
0003
09/09/19 01:23:17 GOMIUC0185L IUCV ACCEPTING SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003 USR=TS3163
0003
09/09/19 01:23:17 GOMIUC0188L IUCV SYSTEM AND USER SET SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003
USR=TS3163 0003
09/09/19 01:23:17 GOMCMD0201L "VIEWSPL SFID MAINT640,068 MODE RDR" SYS=ZVMTESTS VID=TS3163
ALI=_GOM0003 USR=TS3163 SRC=MASIUCV CLS=99
09/09/19 01:23:17 GOMIUC0186L IUCV DISCONNECTING SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003 USR=TS3163
0003
09/09/19 01:24:04 GOMIUC0185L IUCV ACCEPTING SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003 USR=TS3163 0003
09/09/19 01:24:04 GOMIUC0188L IUCV SYSTEM AND USER SET SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003
USR=TS3163 0003
09/09/19 01:24:04 GOMCMD0201L "VIEWSPL SFID MAINT640,068 MODE RDR CLASS A" SYS=ZVMTESTS VID=TS3163
ALI=_GOM0003 USR=TS3163 SRC=MASIUCV CLS=99
09/09/19 01:24:04 GOMIUC0186L IUCV DISCONNECTING SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003 USR=TS3163
0003
09/09/19 01:25:30 GOMIUC0185L IUCV ACCEPTING SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003 USR=TS3163 0003
09/09/19 01:25:30 GOMIUC0188L IUCV SYSTEM AND USER SET SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003
USR=TS3163 0003
09/09/19 01:25:30 GOMCMD0201L "VIEWLOG" SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003 USR=TS3163
SRC=MASIUCV CLS=99
09/09/19 01:25:30 GOMCMD0201L "VIEWLOG" SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003 USR=TS3163
SRC=MASIUCV CLS=99
```

Figure 123. VIEWLOG command without operands

**Example 2: Use the VIEWLOG command to display a specific number of lines (10) from the Operations Manager log.**

```
GOMCMD OPMGRM1 VIEWLOG LINES 10
09/09/19 01:24:04 GOMIUC0188L IUCV SYSTEM AND USER SET SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003
USR=TS3163 0003
09/09/19 01:24:04 GOMCMD0201L "VIEWSPL SFID MAINT640,068 MODE RDR CLASS A" SYS=ZVMTESTS
VID=TS3163 ALI=_GOM0003 USR=TS3163 SRC=MASIUCV CLS=99
09/09/19 01:24:04 GOMIUC0186L IUCV DISCONNECTING SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003
USR=TS3163 0003
09/09/19 01:25:30 GOMIUC0185L IUCV ACCEPTING SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003 USR=TS3163
0003
09/09/19 01:25:30 GOMIUC0188L IUCV SYSTEM AND USER SET SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003
USR=TS3163 0003
09/09/19 01:25:30 GOMCMD0201L "VIEWLOG" SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003 USR=TS3163
SRC=MASIUCV CLS=99
09/09/19 01:32:41 GOMIUC0186L IUCV DISCONNECTING SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003
USR=TS3163 0003
09/09/19 01:35:20 GOMIUC0185L IUCV ACCEPTING SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003 USR=TS3163
0003
09/09/19 01:35:20 GOMIUC0188L IUCV SYSTEM AND USER SET SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003
USR=TS3163 0003
09/09/19 01:35:20 GOMCMD0201L "VIEWLOG LINES 10" SYS=ZVMTESTS VID=TS3163 ALI=_GOM0003
USR=TS3163 SRC=MASIUCV CLS=99
```

Figure 124. VIEWLOG command with LINES operand

**Example 3: Use VIEWLOG with the DATE operand to specify that a copy of the log for a specific date (2019/08/28) is to be sent to virtual reader.**

```
GOMCMD OPMGRM1 VIEWLOG DATE 2019/08/28
RDR FILE 0028 SENT FROM OPMGRM1 PUN WAS 0030 RECS 1524 CPY 001 A NOHOLD NOKEEP
Ready; T=0.01/0.01 06:32:03
```

Figure 125. VIEWLOG command with DATE operand

## Usage notes

1. LINES and DATERNG are mutually exclusive.
2. The date format displayed in VIEWLOG will be based on the CP DATEFORMAT setting of the user issuing VIEWLOG. If MODE RDR or DATE is specified, the date format will be provided in yyyy/mm/dd format.
3. You can use the following default PF KEY assignments to move through the log display:
  - PF01 — Show the help screen.
  - PF02 — Pass the current VIEWLOG session contents to XEDIT.
  - PF03 — End the VIEWLOG session.
  - PF05 — Toggle the acknowledgment for a line in the log.
  - PF07 — Scroll up.
  - PF08 — Scroll down.
  - PF10 — Scroll left.
  - PF11 — Scroll right.
  - PF12 — Retrieve previous command.
  - PF19 — Scroll to the top.
  - PF20 — Scroll to the bottom.
  - PF21 — Process CMS command.
  - PF22 — Display the users viewing the log.
  - PF23 — Toggle auto scrolling.
  - PF24 — Display the PF key assignments.

- PA02 — Toggle auto scrolling.
- ENTER — Process the command on the command line.

**Note:** For the scrolling PF keys, you can enter a scrolling amount on the command line. If no amount is specified, the scroll amount is equal to the size of the viewing area.

A command may be typed on the command line for the process CMS command action.

4. To invoke XEDIT to review the current content, you must have sufficient disk space to **save** the current content. The filename and filetype of the file created by Operations Manager will be `_GOMALOG systemname`. If a file already exists, it will be overwritten.

The default filemode of the file created by Operations Manager is **A1**. The XEDIT rules for filemode processing remain in effect. Thus, using the default filemode A1 will result in the file remaining on the A-disk upon exit.

You may control the file mode used to save the current content by using the `XEDITFM=fm` setting in VIEWLOG PROFILE. For example, by using `XEDITFM=A3`, if you do not save any changes to the file during the XEDIT session, it will be erased upon exit.

Any changes made while using XEDIT are not reflected in the live view upon returning from XEDIT.

5. Issue the following command in the VIEWLOG PROFILE file to control the location, sharing, and disposition of the contents passed to XEDIT:

```
XEDITFM=m[n]
```

where *m* is the file mode *letter* and *n* is the file mode *number*.

- The minidisk/SFS accessed as *m*, must be R/W (read/write).
- The file mode number may be any valid CMS file number. You can specify a file mode number of 3 and the file passed to XEDIT will be erased by XEDIT after reading the file. If you specify a file mode number of 3, you must use the SAVE/FILE XEDIT command to retain the file.

The default is `XEDITFM=A1`.

6. The amount of data displayed or sent to your virtual reader from the log is limited to the size of the logging data space. The *logsize* parameter on GOMMAIN determines this limit. If you want to view the entire log, use the DATE operand to obtain a log of the entire day.
7. To search through the display, use the forward slash (/) to search forward and the back slash (\) to search backward. Use the equal sign (=) to repeat the previous command. See [“DEFOPTN” on page 57](#) for information on how to change these characters.
8. The actions for PF01 - PF24 and initial value for the PF key display are configurable through the VIEWLOG PROFILE file.
  - a. Issue the following command in the VIEWLOG PROFILE file to set the PF key assignments:

```
PFnn=action
```

Where *nn* is 01 through 24 and *action* is a valid PF key action.

The valid PF key actions are:

- HELP — Show the help screen.
- XEDIT — Pass the current VIEWLOG session contents to XEDIT.
- END — End the VIEWLOG session.
- HOLD — Toggle the acknowledgment for a line in the log.
- UP — Scroll up.
- DOWN — Scroll down.
- LEFT — Scroll left.
- RIGHT — Scroll right.

- RECALL — Retrieve previous command.
- TOP — Scroll to top.
- BOT — Scroll to bottom.
- EXCMD — Process a CMS command (the command to process may be included).

Refer to Usage Note “9” on page 164 for more information.

- USERS — Display the users viewing the log.
- SCROLL — Toggle auto scrolling.
- VIEWPF — Display the PF key assignments.

- b. Issue the following command in the VIEWLOG PROFILE file to control the display of the PF key information lines:

```
SHOWPF=c[,h,xat,clr]
```

Where *c* is one of the following options:

- N — Do not display the PF key assignments on the screen.
- Y — Display PF01–PF12 assignments on the screen.
- A — Display PF01–PF24 assignments on the screen.

The values for *h,xat,clr* are optional. If one of the values is specified, they must all be specified.

- *h* — Specifies if the line is highlighted. Values are Y or N.
- *xat* — Specifies the extended attributes applied to the line. Values are:
  - AAI — Leave as is
  - ABL — Blinking
  - ARV — Reverse video
  - AUL — Underline
- *clr* — Specifies the color applied to the line. Values are:
  - CAI — Leave as is
  - CBL — Blue
  - CRE — Red
  - CPI — Pink
  - CGR — Green
  - CCY — Cyan
  - CYE — Yellow
  - CWH — White

The following shows an example of what might be in a VIEWLOG PROFILE file after customizing a PF key assignment and the display of the PF keys:

```
PF13=HELP
SHOWPF=Y,Y,AAI,CAI
```

9. While using VIEWLOG, you can also execute a z/VM or Operations Manager command by using EXCMD. Execute the EXCMD command with one of the following methods:

- Press a function key that is assigned to EXCMD without any command on the command line. This invokes CMS SUBSET. To return to VIEWLOG, use the CMS command `return`.
- Press a function key that is assigned to EXCMD with a CMS command on the command line.

When the command completes, if the command on the command line is not an Operations Manager command, you return to the originating VIEWLOG command.

Assign EXCMD with a z/VM or Operations Manager command to a PF key in VIEWLOG PROFILE. For example, PF19=EXCMD LISTFILE \* \* A or PF20=EXCMD GOMCMD OPMGRM1 VIEWSPPL.

When the command completes, if the command assigned to the PF key is an Operations Manager command, you return to the point that originated the VIEWLOG command.

DEFOPTN  
GOMMAIN

Table 105 on page 165 lists the return codes for the VIEWLOG command.

Return code	Description
0	Finished correctly
-100	Error

```

graph LR
    subgraph Path_A [PATH A]
        A1[VIEWSP] --> A2[ ]
    end
    subgraph Path_B [PATH B]
        B1[SFID userid, splnbr] --> B2[ ]
        B2 --> B3[MODERDR]
        B3 --> B4[ ]
        B4 --> B5[ ]
        B4 --> B6[ ]
    end
    B4 --> B7[Class A]
    B4 --> B8[Class cls]
    B7 --> B9[ ]
    B8 --> B9
    B9 --> B10[ ]
  
```

**PATH A**

VIEWSP

**PATH B**

SFID *userid, splnbr*

MODERDR

Class — A

Class — *cls*

## Authorization

## Related tasks

**Related reference**  
AUTH

# Purpose

Use the VIEWSPPL command to display a list of spool files, view the contents of a spool file in full-screen mode, or place a copy of the contents of a spool file in your virtual reader. .

# Operands

## USER *userid*

Specifies the *userid* whose spool files you want to display. The *userid* can be a wildcard pattern. The wildcard pattern rules are explained in [Appendix B, “Wildcard pattern matching,” on page 181](#).

## DATE *yyyy/mm/dd*

Specifies the earliest creation date of the spool files you want to display. Use the format *yyyy/mm/dd*, where:

- *yyyy*. The year, which must be specified with four digits.
- *mm*. The month (01-12).
- *dd*. The day of the month (01-31).

## SIZE *size*

Specifies the minimum size of the spool files to display. The size must be a number in bytes or suffixed with K for kilobytes, M for megabytes, or G for gigabytes.

## SVOL *vol,vol,...,vol*

Specifies the DASD volume serial number(s) of the spool volume(s) for selecting spool files. You can specify up to eight volumes. The *vol* value can be a wildcard pattern.

## SFID *userid,splnbr*

Specifies that you want to view the spool file contents in full-screen mode and skip viewing the list of spool files. If SFID is used with the MODE RDR operand, a copy of the spool file contents are placed in your virtual reader. The *userid* cannot contain wildcards. The *splnbr* is the 4-digit spool file number. If the spool file number contains leading zeros, you can abbreviate the number. For example, if the spool file number is 0003, you can specify 3.

## MODE RDR

Specifies that a copy of the current spool file is to be placed on your virtual reader.

## Class *cls*

Specifies the reader file class. The default is class A.

# Operand value lengths

Table 106 on page 166 lists the operand value lengths for the VIEWSPPL command.

Table 106. VIEWSPPL operand value lengths	
Operand	Length
CLASS	1
DATE	10
MODE	3-8
SFID	3-13
SIZE	1-10
SVOL	1-55
USER	1-8

## Examples

The following examples show how to use the VIEWSPL command. The examples use different or no operands and display sample results.

### Example 1: Use VIEWSPL to display a list of all users and their associated spool files.

An example of the command is shown below:

GOMCMD OPMGRM1 VIEWSPL											
System: ZVMTESTS		Spool:		1% Used		Files:		0% Used		1 of 484	
1655640		Max:		6.9G		Max:					
Cmd	Owner	File	CLS	QUE	TYP	Size	Hold	Date	Time	Name	Type
	MAINT640	0058	T	RDR	CON	4K	NONE	07/30/19	14:42	SERVICE	\$CONXCLO
	MAINT640	0004	T	RDR	CON	4K	NONE	07/30/19	08:40		
	MAINT640	0063	T	RDR	CON	4K	NONE	07/30/19	15:01		
	OP1	0001	T	RDR	CON	4K	NONE	07/30/19	08:34		
	BLDSEG	0008	T	RDR	CON	4K	NONE	07/30/19	14:58		
	MAINT640	0015	T	RDR	CON	4K	NONE	07/30/19	08:44		
	MAINT640	0067	T	RDR	CON	4K	NONE	07/30/19	15:09		
	MAINT640	0068	T	RDR	CON	4K	NONE	07/30/19	15:08		
	RACFVM	0006	T	RDR	CON	4K	NONE	07/30/19	15:08		
	RACFVM	0005	T	RDR	CON	4K	NONE	07/30/19	15:08		
	MAINT640	0066	T	RDR	CON	4K	NONE	07/30/19	15:08		
	MAINT640	0065	T	RDR	CON	4K	NONE	07/30/19	15:03		
	MAINT640	0061	T	RDR	CON	4K	NONE	07/30/19	14:54		
	MAINT	0017	T	RDR	CON	8K	NONE	07/30/19	15:03		
	MAINT	0018	T	RDR	CON	8K	NONE	07/30/19	15:03		
	MAINT	0019	T	RDR	CON	8K	NONE	07/30/19	15:03		
	MAINT	0020	T	RDR	CON	8K	NONE	07/30/19	15:03		
	RACFVM	0004	T	RDR	CON	4K	NONE	07/30/19	15:03		
	RACFVM	0003	T	RDR	CON	4K	NONE	07/30/19	15:02		
	MAINT640	0059	T	RDR	CON	4K	NONE	07/30/19	14:53		
PRESS F3											
Ready; T=0.01/0.01 01:56:13											

Figure 127. VIEWSPL with no operands

### Example 2: Use VIEWSPL to display a list of spool files for a specific user, MAINT640.

An example of the command is shown below:

GOMCMD OPMGRM1 VIEWSPL USER MAINT640										
--------------------------------------	--	--	--	--	--	--	--	--	--	--

```

System: ZVMTESTS   Spool:   1% Used   Files:   0% Used   1 of 326
                   Max:    6.9G      Max: 1655640

Cmd  Owner      File CLS QUE TYP  Size Hold Date    Time    Name      Type
MAINT640 0058 T RDR CON  4K NONE 07/30/19 14:42
MAINT640 0004 T RDR CON  4K NONE 07/30/19 08:40  SERVICE  $CONXCLED
MAINT640 0063 T RDR CON  4K NONE 07/30/19 15:01
MAINT640 0015 T RDR CON  4K NONE 07/30/19 08:44
MAINT640 0067 T RDR CON  4K NONE 07/30/19 15:09
MAINT640 0068 T RDR CON  4K NONE 07/30/19 15:08
MAINT640 0066 T RDR CON  4K NONE 07/30/19 15:08
MAINT640 0065 T RDR CON  4K NONE 07/30/19 15:03
MAINT640 0061 T RDR CON  4K NONE 07/30/19 14:54
MAINT640 0059 T RDR CON  4K NONE 07/30/19 14:53
MAINT640 0038 T RDR CON  4K NONE 07/30/19 13:34
MAINT640 0034 T RDR CON  4K NONE 07/30/19 13:33  SERVICE  $CONXCLED
MAINT640 0050 T RDR CON  8K NONE 07/30/19 14:32  SERVICE  $CONXCLED
MAINT640 0049 T RDR CON  4K NONE 07/30/19 14:29
MAINT640 0046 A RDR PUN 16K NONE 07/30/19 13:53  DIRMAINT NEWMAIL
MAINT640 0045 T RDR CON 28K NONE 07/30/19 13:53
MAINT640 0044 T RDR CON  4K NONE 07/30/19 13:52
MAINT640 0043 T RDR CON  4K NONE 07/30/19 13:51
MAINT640 0041 T RDR CON  4K NONE 07/30/19 13:51
MAINT640 0039 T RDR CON 12K NONE 07/30/19 13:40

PRESS F3
Ready; T=0.01/0.01 02:01:30

```

Figure 128. VIEWSPPL command with USER operand

**Example 3: Use VIEWSPPL to display the earliest creation date (or later) of spool files you want to display.**

An example of the command is shown below:

```
GOMCMD OPMGRM1 VIEWSPPL DATE 2019/08/28
```

**Note:** The result shown below does not list any files with the exact specified date, but shows others created after that date.

```

System: ZVMTESTS   Spool:   1% Used   Files:   0% Used   1 of 14
                   Max:    6.9G      Max: 1655640

Cmd  Owner      File CLS QUE TYP  Size Hold Date    Time    Name      Type
MAINT640 0463 T RDR CON 12K NONE 09/05/19 01:55
MAINT640 0464 T RDR CON  4K NONE 09/05/19 06:01
TS3163  0021 A RDR PUN  2M NONE 09/06/19 01:23  ZVMTESTS 20190905
TS3163  0022 A RDR PUN  2M NONE 09/06/19 01:23  ZVMTESTS 20190905
TS3163  0023 A RDR PUN  1M NONE 09/06/19 01:24  ZVMTESTS 20190904
TS3163  0024 A RDR PUN  96K NONE 09/06/19 01:24  ZVMTESTS 20190826
TS3163  0025 A RDR PUN 156K NONE 09/06/19 01:24  ZVMTESTS 20190822
MAINT640 0466 A RDR PUN  20K NONE 09/08/19 11:48  ZVMTESTS 20190908
DIRMAINT *0129 3 PRT CON 144K USER 09/05/19 00:01
DIRMAINT *0130 2 PRT CON 144K USER 09/06/19 00:01
MAINT640 *0465 T PRT CON  12K NONE 09/06/19 04:15
DIRMAINT *0131 1 PRT CON 144K USER 09/07/19 00:01
DIRMAINT *0132 0 PRT CON 144K USER 09/08/19 00:01
DIRMAINT *0133 0 PRT CON  64K USER 09/09/19 00:01

```

Figure 129. VIEWSPPL command with DATE operand

**Example 4: Use VIEWSPPL to display the spool files that are the specified size, 144K, or larger.**

An example of the command is shown below:

```
GOMCMD OPMGRM1 VIEWSPPL SIZE 144K
```



System: ZVMTESTS		Spool: 1% Used		Files: 0% Used		1 of 20					
		Max: 6.9G		Max: 1655640							
Cmd	Owner	File	CLS	QUE	TYP	Size	Hold	Date	Time	Name	Type
	TS3163	0003	A	RDR	PUN	252K	NONE	08/26/19	05:26	ZVMTESTS	20190823
	TS3163	0005	A	RDR	PUN	252K	NONE	08/26/19	05:30	ZVMTESTS	20190823
	TS3163	0020	A	RDR	PUN	156K	NONE	08/28/19	06:32	ZVMTESTS	20190828
	TS3163	0021	A	RDR	PUN	2M	NONE	09/06/19	01:23	ZVMTESTS	20190905
	TS3163	0022	A	RDR	PUN	2M	NONE	09/06/19	01:23	ZVMTESTS	20190905
	TS3163	0023	A	RDR	PUN	1M	NONE	09/06/19	01:24	ZVMTESTS	20190904
	TS3163	0025	A	RDR	PUN	156K	NONE	09/06/19	01:24	ZVMTESTS	20190822
	OPERATOR	*0007	T	PRT	CON	908K	NONE	07/30/19	16:13		
	RACFVM	*0008	T	PRT	CON	192K	NONE	07/30/19	16:13		
	TCPIP	*0003	T	PRT	CON	168K	NONE	08/21/19	16:36		
	DIRMAINT	*0123	9	PRT	CON	144K	USER	08/30/19	00:01		
	DIRMAINT	*0124	8	PRT	CON	144K	USER	08/31/19	00:01		
	DIRMAINT	*0125	7	PRT	CON	144K	USER	09/01/19	00:01		
	DIRMAINT	*0126	6	PRT	CON	144K	USER	09/02/19	00:01		
	DIRMAINT	*0127	5	PRT	CON	144K	USER	09/03/19	00:01		
	DIRMAINT	*0128	4	PRT	CON	144K	USER	09/04/19	00:01		
	DIRMAINT	*0129	3	PRT	CON	144K	USER	09/05/19	00:01		
	DIRMAINT	*0130	2	PRT	CON	144K	USER	09/06/19	00:01		
	DIRMAINT	*0131	1	PRT	CON	144K	USER	09/07/19	00:01		
	DIRMAINT	*0132	0	PRT	CON	144K	USER	09/08/19	00:01		

Figure 130. VIEWSPPL command with SIZE operand

**Example 5: Use the VIEWSPPL command to display the contents of a specific spool file, 0068, for a specific user, MAINT640.**

An example of the command is shown below:

```
GOMCMD OPMGRM1 VIEWSPPL SFID MAINT640,0068
```

Sample results are shown below.

```

MAINT640 0068                                Block: 1 of 1
RPIMGR031E RESOURCE MAINT640 SPECIFIED BY SPOOL COMMAND NOT FOUND
ICH70001I VMSERVS  LAST ACCESS AT 15:03:01 ON TUESDAY, JULY 30, 2019
ICH70001I VMSERVU  LAST ACCESS AT 15:03:01 ON TUESDAY, JULY 30, 2019
AUTO LOGON ***          VMSERVS  USERS = 8
HCPCLS6056I XAUTOLOG information for VMSERVS: The IPL command is verified by the IPL command
processor.
ICH70001I VMSERVR  LAST ACCESS AT 15:03:01 ON TUESDAY, JULY 30, 2019
AUTO LOGON ***          VMSERVU  USERS = 9
ICH70001I VMSERVP  LAST ACCESS AT 15:03:01 ON TUESDAY, JULY 30, 2019
HCPCLS6056I XAUTOLOG information for VMSERVU: The IPL command is verified by the IPL command
processor.
AUTO LOGON ***          VMSERVR  USERS = 10
HCPCLS6056I XAUTOLOG information for VMSERVR: The IPL command is verified by the IPL command
processor.
ICH70001I DTCVSW1  LAST ACCESS AT 15:03:01 ON TUESDAY, JULY 30, 2019
ICH70001I DTCVSW2  LAST ACCESS AT 15:03:01 ON TUESDAY, JULY 30, 2019
ICH70001I DTCVSW3  LAST ACCESS AT 15:03:01 ON TUESDAY, JULY 30, 2019
AUTO LOGON ***          VMSERVP  USERS = 11
ICH70001I DTCVSW4  LAST ACCESS AT 15:03:01 ON TUESDAY, JULY 30, 2019
HCPCLS6056I XAUTOLOG information for VMSERVP: The IPL command is verified by the IPL command
processor.
AUTO LOGON ***          DTCVSW1  USERS = 12
HCPCLS6056I XAUTOLOG information for DTCVSW1: The IPL command is verified by the IPL command
processor.
ICH70001I TCPIP    LAST ACCESS AT 14:08:11 ON TUESDAY, JULY 30, 2019
AUTO LOGON ***          DTCVSW2  USERS = 13
HCPCLS6056I XAUTOLOG information for DTCVSW2: The IPL command is verified by the IPL command
processor.
AUTO LOGON ***          DTCVSW3  USERS = 14
HCPCLS6056I XAUTOLOG information for DTCVSW3: The IPL command is verified by the IPL command
processor.
AUTO LOGON ***          DTCVSW4  USERS = 15
AUTO LOGON ***          TCPIP     USERS = 16
HCPCLS6056I XAUTOLOG information for TCPIP: The IPL command is verified by the IPL command
processor.
HCPCLS6056I XAUTOLOG information for DTCVSW4: The IPL command is verified by the IPL command
processor.
ICH70001I DIRMAINT LAST ACCESS AT 15:03:31 ON TUESDAY, JULY 30, 2019
Exiting AUTOLOG1 profile exec with return code =0
Logoff AUTOLOG1
LOGOFF AT 15:09:21 CDT TUESDAY 07/30/19

```

Figure 131. VIEWSPL command with SFID USER,SPLNBR operands

**Example 6: Use the VIEWSPL command to copy the specified spool file (0068) for this user (MAINT640) and send it to the virtual reader (MODE RDR).**

An example of the command is shown below:

```
GOMCMD OPMGRM1 VIEWSPL SFID MAINT640,0068 MODE RDR
```

An example of the results is shown below:

```

GOMCMD OPMGRM1 VIEWSPL SFID MAINT640,068 MODE RDR
Ready; T=0.01/0.01 06:15:07
RDR FILE 0026 SENT FROM OPMGRM1  PRT WAS 0028 RECS 0032 CPY  001 A NOHOLD NOKEEP

```

Figure 132. VIEWSPL command with MODE RDR operand

**Example 7: Uses the VIEWSPL command to identify a specific spool file number (0068), user ID (MAINT640), virtual reader, (MODE RDR) and assigned reader file class (CLASS A).**

An example of the command is shown below:

```
GOMCMD OPMGRM1 VIEWSPL SFID MAINT640,0068 MODE RDR CLASS A
```

An example of the results is shown below:

```
GOMCMD OPMGRM1 VIEWSPL SFID MAINT640,068 MODE RDR CLASS
A
RDR FILE 0027 SENT FROM OPMGRM1 PRT WAS 0029 RECS 0032 CPY 001 A NOHOLD NOKEEP
Ready; T=0.01/0.01 06:18:20
```

Figure 133. VIEWSPL command with several operands

## Usage notes

Additional information about the VIEWSPL command is provided below.

### General usage notes

VIEWSPL can be used to sort the output by a specific column. When the output is sorted by one of the columns, the column heading is highlighted.

The date format displayed in VIEWSPL will be based on the CP DATEFORMAT setting of the user that issued VIEWSPL. If MODE(RDR) is specified, the date format will be provided in yyyy/mm/dd format.

See Appendix B, “Wildcard pattern matching,” on page 181 for an explanation of wildcard pattern matching.

### Viewing spool file list

1. You can use the following default PF keys to move through the displayed spool file list:

- PF01 — Show the help screen
- PF02 — View the spool file contents
- PF03 — End the VIEWSPL session
- PF04 — Toggle the display format.
- PF05 — Sort the files in ascending order
- PF06 — Sort the files in descending order
- PF07 — Scroll up
- PF08 — Scroll down
- PF10 — Scroll left
- PF11 — Scroll right
- PF19 — Scroll to top
- PF20 — Scroll to bottom
- PF21 — Process CMS command
- PF24 — Display the PF key assignments
- ENTER — Process the command in the command field

2. The list of spool files includes:

- files allowed by the \*SPL GENERIC interface
- PRT files
- RDR files
- PUN files
- DMP files
- HELD files (USER or SYS HOLD)
- OPEN files

3. The list of spool files does not include:

- HELP files

- SYS files (DMP, NSS, NLS, and so on)

4. Available CP commands in the spool file list:

- Issue the PURGE, CHANGE, and TRANSFER commands to manipulate the spool files. Enter these commands followed by a forward slash (/) and the command operands. Refer to the *CP Commands and Utilities Reference (SC24-6081)* for more information and the valid abbreviations for these commands. If an asterisk (\*) is encountered within the command operands, it is modified to contain the USERID executing the VIEWSPPL command. The processing of the asterisk (\*) was changed in PTF UI52532.
- To repeat the previous command, type the equal sign (=) in the command field. To change the command symbols, see “DEFOPTN” on page 57.
- To sort a column, place the cursor on the column heading and use the PF05 or PF06 key.
- The number of spool files selected is limited to 25,000 files.
- The value for space utilization percentage is obtained by rounding downward. This value may differ from the space utilization percentage computed from other sources.

5. The spool file list can contain the following indicators before the spool file number:

Table 107. Spool file list indicators	
Indicator	Description
' '	Local file that can be viewed
'*'	Local file that cannot be viewed using the PF02 key, or whichever key is assigned the VIEW command from the VIEWSPPL PROFILE
'+'	Non-local file that can be viewed
'*+'	Non local file that cannot be viewed using the PF02 key, or whichever key is assigned the VIEW command from the VIEWSPPL PROFILE

A file cannot be viewed for one of the following reasons:

- the file is HELD (SYS and/or USER)
- the file was created by CP (DMP, and so on)
- the file is currently open and in use

The following screen is an example of a spool file list, where file 0145 is local and viewable, file 0029 is local and non-viewable, file 0142 is non-local and viewable, and file 0082 is non-local and non-viewable:

System: RS54		Spool: 55% Used		Files: 0% Used		196 of 202					
		Max: 2.4G		Max: 1655640							
Cmd	Owner	File	CLS	QUE	TYP	Size	Hold	Date	Time	Name	Type
	MAINT710	0145	T	RDR	CON	4K	NONE	10/26/2012	17:52		
	OPERATOR	*0029	T	PRT	CON	88K	NONE	12/05/2012	18:41		
	MAINT710	+0142	A	PRT	PRT	4K	NONE	05/08/2013	13:20		
	RVBACKUP	*+0082	T	PRT	CON	1M	USER	05/24/2013	13:32		

6. While using VIEWSPPL, you can also execute a z/VM or Operations Manager command by using EXCMD. Execute the EXCMD command with one of the following methods:

- Press a function key that is assigned to EXCMD without any command on the command line. This invokes CMS SUBSET. To return to VIEWSPPL, use the CMS command `return`.
- Press a function key that is assigned to EXCMD with a CMS command on the command line.

When the command completes, if the command on the command line is not an Operations Manager command, you return to the originating VIEWSPPL command.

When the command completes, if the command on the command line is an Operations Manager command, you return to the point that originated the VIEWSPPL command.

- Assign EXCMD with a z/VM or Operations Manager command to a PF key in VIEWSPPL PROFILE. For example, PF19=EXCMD LISTFILE \* \* A or PF20=EXCMD GOMCMD OPMGRM1 VIEWCON USER(OPERATOR).

When the command completes, if the command assigned to the PF key is not an Operations Manager command, you return to the originating VIEWSPPL command.

When the command completes, if the command assigned to the PF key is an Operations Manager command, you return to the point that originated the VIEWSPPL command.

7. The value for space utilization percentage is obtained by rounding downward. This value may differ from the space utilization percentage computed from other sources.
8. The spool file list can contain the '+' indicator following the volume. This indicates the spool file allocation includes additional volume(s). Use the scroll right function (PF11) to display the additional volumes. A particular spool file can have data allocation on up to 250 volumes. The scroll right function (PF11) can be used to scroll through all volumes that contain data for a spool file.
9. For the VIEWSPPL command to display and allow selection of a spool file's allocation information, install the updates to z/VM that contain the support. The z/VM support is available in z/VM V7.2 with PTF UM35825.

## Viewing spool file contents

You can use the following default PF keys to move through the displayed spool file contents:

- PF01 — Show the help screen
- PF03 — Return to the spool file list
- PF07 — Scroll up
- PF08 — Scroll down
- PF10 — Scroll left
- PF11 — Scroll right
- PF19 — Scroll to top
- PF20 — Scroll to bottom
- PF21 — Process CMS command
- PF24 — Display the PF key assignments
- ENTER — Process the command in the command field

Consider the following when using these PF keys:

- A command may be typed on the command line for the process CMS command action.
- When viewing a spool file, you can search through the display. Use the forward slash (/) to search forward and the back slash (\) to search backward. Use the equal sign (=) to repeat the previous command. Issue the DEFOPTN command to change these characters. These characters are only valid while viewing the contents of a file. They are not valid and will not work while viewing a list of spool files. To change the command symbols, refer to [“DEFOPTN” on page 57](#).
- When viewing the contents of a spool file, the VIEW, SORTA, and SORTD action keys have no effect. They are considered null PF keys.

## Setting values through the profile

The actions for PF01-PF24, initial value for the sort order, and display of the PF key information lines are configurable through the VIEWSPPL PROFILE file.

1. Issue the following command in the VIEWSPPL PROFILE file to customize the actions for PF01 - PF24:

*PFnn=action*

Where *nn* is 01 through 24 and *action* is a valid PF key action.

The valid PF key actions while viewing the spool file list are:

- HELP — Show the help screen
- VIEW — View the spool file contents
- END — End the VIEWSPPL session
- SORTA — Sort the files in ascending order
- SORTD — Sort the files in descending order
- UP — Scroll up
- DOWN — Scroll down
- LEFT — Scroll left
- RIGHT — Scroll right
- TOP — Scroll to top
- BOT — Scroll to bottom
- VIEWPF — Display the PF key assignments
- FORMAT — Toggle the display format.

Refer to Note “4” on page 175 for more information.

- EXCMD — Execute a z/VM or Operations Manager command

Refer to Usage Note “6” on page 172 for more information.

The valid PF key actions while viewing the spool file content are:

- HELP — Show the help screen
- END — Return to the spool file list
- UP — Scroll up
- DOWN — Scroll down
- LEFT — Scroll left
- RIGHT — Scroll right
- TOP — Scroll to top
- BOT — Scroll to bottom
- VIEWPF — Display the PF key assignments
- EXCMD — Execute a z/VM or Operations Manager command

Refer to Usage Note “6” on page 172 for more information.

2. Issue the SORT command in the VIEWSPPL PROFILE file to customize the initial sort order of the spool files. Only one column can be specified. You can sort a column in ascending (A) or descending (D) order. If no sort order is provided, the default is ascending (A). Use the following format:

`SORT=column[,A|D]`

The columns you can sort are:

- OWNER
- FILE
- CLS
- QUE
- TYP
- SIZE
- HOLD
- DATE

- NAME
- TYPE
- DIST
- DEST
- COPY
- KEEP
- MSG
- USERFORM
- OPERFORM
- FCB

3. Issue the following command in the VIEWSPL PROFILE file to control the display of the PF key information lines:

```
SHOWPF=c[,h,xat,clr]
```

Where *c* is one of the following options:

- N — Do not display the PF key assignments on the screen.
- Y — Display PF01–PF12 assignments on the screen.
- A — Display PF01–PF24 assignments on the screen.

The values for *h,xat,clr* are optional. If one of the values is specified, they must all be specified.

- *h* — Specifies if the line is highlighted. Values are Y or N.
- *xat* — Specifies the extended attributes applied to the line. Values are:
  - AAI — Leave as is
  - ABL — Blinking
  - ARV — Reverse video
  - AUL — Underline
- *clr* — Specifies the color applied to the line. Values are:
  - CAI — Leave as is
  - CBL — Blue
  - CRE — Red
  - CPI — Pink
  - CGR — Green
  - CCY — Cyan
  - CYE — Yellow
  - CWH — White

The following shows an example of what might be in a VIEWSPL PROFILE file after customizing the initial sort order of the spool files and the display of the PF keys:

```
PF13=HELP
SORT=OWNER,A
SHOWPF=Y,Y,AAI,CAI
```

4. Issue one of the following commands in the VIEWSPL PROFILE file to toggle the display format:

- FORMAT=0 — Displays the spool information with allocation information located on horizontally scrolling panels 3-n.
- FORMAT=1 — Displays the spool information with allocation information located on horizontally scrolling panels 1 and 3-n.

## Return codes

Table 108 on page 176 lists the return codes for the VIEWSPPL command.

Table 108. VIEWSPPL return codes	
Return code	Description
0	Finished correctly
-100	Error



---

## Appendix A. Sample files

Several sample files are shipped with Operations Manager. These samples help you create customized statements or EXEC files for your site. The files are located on the 5697J10F 2C2 minidisk or associated SFS directory.

### Description of sample files

#### OPMGRM1 CONFSAMP

Contains sample configuration statements for defining the Operations Manager environment and who is authorized to use it. The OPMGRM1 CONFSAMP sample statements include the following actions for use of Operations Manager:

- Define a group and member(s).
- Authorize the SYSADM group for all Operations Manager functions.
- Define users with config authority to Operations Manager.
- Define users and the consoles they can access.
- Define users and the spool files they can access.
- Define action processing servers.
- Imbed a configuration file specific to this z/VM system.

Refer to [Chapter 2, “Configuring Operations Manager,”](#) on page 5 for more information on how to use this sample to create the Operations Manager startup configuration file.

#### EXAMPLES CONFSAMP

Contains sample Operations Manager configuration statements to do the following:

- Perform some initial monitoring of spool space, page space, console messages, minidisk full conditions, events, and logoff status of critical user IDs. Alerts can be sent via email or to a z/VM user ID that is used as a central operations console.
- Schedule spool cleanup.
- Schedule expiration processing in IBM Backup and Restore Manager for z/VM and IBM Archive Manager for z/VM.
- Request that all Operations Manager servers re-access the OPMGRM1 198 disk where configuration information and, possibly, your Rexx EXECs are stored.
- Perform actions on remote Operations Manager systems.
- Send data to a remote syslog server/daemon.
- Send a message to syslog.
- Suppress messages based on User ID or guest console.
- Schedule regular checks to monitor dump space.

To use the samples in EXAMPLES CONFSAMP, you must do one of the following:

- Copy the example(s) to your startup configuration file for Operations Manager, or
- Copy the file to the OPMGRM1 198 disk, rename the file, and add a CONFIG statement to the startup configuration file that references this additional configuration file.

In addition, the examples must be modified to work in your environment. It is required to update the email address and the user ID for the central operations console. Refer to the comments in EXAMPLES CONFSAMP for more details.

### **CHKDISK SAMPEXEC**

Contains a sample Rexx EXEC for checking the percent full of a specified minidisk. If the percent full is above a specified threshold, call the sample SENDMAIL action in Operations Manager to email someone. Refer to the comments in the sample EXEC for details.

To use this sample EXEC, copy the file to the OPMGRM1 198 disk (or other minidisk accessible to all Operations Manager service machines) and rename it to CHKDISK EXEC. Be sure to review this EXEC and modify it, if necessary, to work in your environment.

### **CHKSFS SAMPEXEC**

Contains a sample Rexx EXEC for checking the percent full of a specified SFS file space. If the percent full is above a specified threshold, call the sample SENDMAIL action in Operations Manager to email someone. Refer to the comments in the sample EXEC for details. To use this sample EXEC, copy the file to the OPMGRM1 198 disk (or other minidisk accessible to all Operations Manager service machines) and rename it to CHKSFS EXEC. Be sure to review this EXEC and modify it, if necessary, to work in your environment.

### **CHKDUMP SAMPEXEC**

Contains a sample Rexx EXEC to monitor dump space. This EXEC is triggered by a schedule in Operations Manager. It issues the **QUERY DUMP** command. If there is insufficient dump space, call the sample SENDMAIL action in Operations Manager to email someone. Refer to the comments in the sample EXEC for details.

To use this sample EXEC, copy the file to the OPMGRM1 198 disk (or other minidisk accessible to all Operations Manager service machines) and rename it to CHKDUMP EXEC. Be sure to review this EXEC and modify it, if necessary, to work in your environment.

### **EBC2ASC SAMPEXEC**

Contains a sample Rexx EXEC for translating from EBCDIC to ASCII using the specified code pages. Refer to the comments in the sample EXEC for details.

To use this sample EXEC, copy the file to the OPMGRM1 198 disk (or other minidisk accessible to all Operations Manager service machines) and rename it to EBC2ASC EXEC. Be sure to review this EXEC and modify it, if necessary, to work in your environment.

### **SENDSYSL.SAMPEXEC**

Contains a sample Rexx EXEC for sending a system log. Refer to the comments in the sample EXEC for details.

To use this sample EXEC, copy the file to the OPMGRM1 198 disk (or other minidisk accessible to all Operations Manager service machines) and rename it to SENDSYSL EXEC. Be sure to review this EXEC and modify it, if necessary, to work in your environment.

### **SMTPMMSG SAMPEXEC**

Contains a sample Rexx EXEC for sending an email notification when an alert condition occurs. Refer to the comments in the sample EXEC for details.

To use this sample EXEC, copy the file to the OPMGRM1 198 disk (or other minidisk accessible to all Operations Manager service machines) and rename it to SMTPMMSG EXEC. Be sure to review this EXEC and modify it, if necessary, to work in your environment.

### **GETSPCON SAMPEXEC**

GETSPCON EXEC is a sample Rexx EXEC that allows you to capture spooled console data and add it to the corresponding service machine's console data in Operations Manager, making it visible in the log and also viewable with the VIEWCON and VIEWLOG commands. This can help you capture missed console data during a system IPL when one or more user IDs and service machines are started before Operations Manager. By using this EXEC, you can retrieve the spooled console data from each of these user IDs and add it to the captured consoles in Operations Manager.

You would normally run this EXEC once per IPL and within a few minutes after IPL. This EXEC can be used for any user ID, including the following IDs, which are typically started before Operations Manager during a system IPL:

- OPERATOR
- RACFVM
- TCPIP

### Recommendation to run this EXEC

It is best to create a new z/VM user ID which will have the required privileges to run this EXEC. You cannot run this EXEC with a user ID that is already using \*MSG (or the STARMMSG or STARSYS stage in PIPELINES), which means you cannot run this EXEC on any of the Operations Manager service machines.

You can place this EXEC on the 191 disk of this user ID and call it from the PROFILE EXEC. (See the GETSPCON DIRECT example below.)

To run the EXEC, you can issue **XAUTOLOG GETSPCON**. This command can be defined as an action in the Operations Manager configuration file so whenever OPMGRM1 is started, the user ID, GETSPCON, is started and the GETSPCON EXEC is run.

**Note:** The GETSPCON EXEC will only capture spooled console data if it is within a specified number of minutes after a system IPL.

The following sample statements can be added to one of your Operations Manager configuration files:

1. Authorize the new user ID GETSPCON to issue DEFIPCS commands.

```
AUTH USER GETSPCON ACCESS CONTROL
```

2. Use XAUTOLOG for the user ID, GETSPCON, so it will run the GETSPCON EXEC.

```
'CP CMD XAUTOLOG GETSPCON'
```

### Sample GETSPCON DIRECT

```
* * * Top of File * * *
USER GETSPCON RACFPW 256M 512M CGD
*
* GETSPCON - get spool console after IPL.
*
  ACCOUNT GOM160 GOM160
  IPL CMS
  MACHINE ESA
  CONSOLE 0009 3215 T OPMGRM1
  SPOOL 000C 2540 READER *
  SPOOL 000D 2540 PUNCH A
  SPOOL 000E 1403 A
*
  LINK MAINT 0190 0190 RR
  LINK MAINT 019E 019E RR
*
  MDISK 0191 3390 7233 5 6K6CM1 MR RRRACF WRRACF MWRACF
* * * End of File * * *
```

### Sample PROFILE EXEC for user ID GETSPCON

```
/*****
/*  PROFILE EXEC FOR GETSPCON                               */
/*****
Address Command
'EXEC GETSPCON OPERATOR'
'CP SLEEP 5 SEC'
'EXEC GETSPCON RACFVM'
'CP SLEEP 5 SEC'
'EXEC GETSPCON TCPIP'
```



---

## Appendix B. Wildcard pattern matching

Wildcard pattern matching is used by several of the commands. Wildcards allow you to search for and process characters that are similar.

For example, instead of searching for STAR1, STAR20, STAR3000, you could type STAR\* (an asterisk) to select all of the items that begin with "STAR". Various wildcard characters are used for selecting different types of text or numbers.

Wildcard pattern matching is performed using the following:

- An asterisk (\*) means zero (0) or more characters of any kind.
- A percent sign (%) means one (1) required character of any kind.
- A pound sign (#) means one (1) required numeric character (0-9).
- An ampersand (&) means one (1) required alphabetic character (A-Z,a-z).
- An "at" sign (@) means one (1) required hexadecimal character (0-9,A-F, a-f).
- A quote mark (") is the escape character for matching any of the above characters.

Wildcard patterns can be used in any combination. If a pattern contains blanks the value may be entered 'pattern'.



---

## Appendix C. Using substitution variables in commands

### Common substitution variables

The substitution variables that can be defined are listed in the table below.

Table 109. Substitution variables used by commands	
Substitution Variable	Description
&U	Substitutes the user/console id of the message text originator.
&R	Substitutes the id of the rule, schedule, or monitor which caused action processing.
&P	Substitutes the value passed by the PARM operand of a rule, schedule, or monitor. The PARM field of the rule, schedule, or monitor may contain substitution variable patterns.
&SYSNAME	Substitutes the system name.
&SYSSIN	Substitutes the system SSI name.
&GOMMAIN	Substitutes the userid of the main processing server.

The following tables list the substitution variables available when an action is triggered specifically by a rule, schedule, and monitor. These substitution variables are available only when the action is triggered by the specified rule, schedule, or monitor statement.

### DEFEMON substitution variables

The DEFEMON command allows the substitution variables listed in the table below.

Table 110. DEFEMON substitution variables	
Variable	Value
&C	0
&T	Concatenation of &0 through &n
&U	Constant _GOMEMON

Table 110. DEFEMON substitution variables (continued)

Variable	Value
&0-&99	<p>Specifies information about the event monitor and event. Valid values are:</p> <ul style="list-style-type: none"> <li>• &amp;0 — Constant "VMEVENT" (Length 7, Type Character)</li> <li>• &amp;1 — Name of the event monitor (Length 1-8, Type Character)</li> <li>• &amp;2 — Name of the event monitor's action (Length 1-8, Type Character)</li> <li>• &amp;3 — Userid that caused the event (Length 1-8, Type Character)</li> </ul> <p><b>Note:</b> In some cases, &amp;3 represents something different. Any differences are listed in the next section, "DEFEMON variables &amp;6-&amp;nn".</p> <ul style="list-style-type: none"> <li>• &amp;4 — Event class (Length 1-8, Type Decimal)</li> <li>• &amp;5 — Event type (Length 1-8, Type Decimal)</li> <li>• &amp;6-&amp;n — Any additional information passed for a particular event class/type. Depending on the message, these variables could include one or more of the following: <ul style="list-style-type: none"> <li>– Reason</li> <li>– Interval specified or interval remaining</li> <li>– Destination or source system</li> <li>– SSI member name and the previous, current, or new SSI mode</li> <li>– Port group name, number, status, or status reason</li> <li>– RDEV number or error status</li> <li>– VSWITCH status</li> <li>– Bridge port status or error status</li> <li>– Other event data</li> </ul> </li> </ul> <p>For more specific parameter details, see the additional DEFEMON information in the section that follows.</p> <p>For more information on event class (&amp;4) and event type (&amp;5), refer to the VM Event System Service (*VMEVENT) described in the <i>CP Programming Services</i> documentation (SC24-6179).</p>

Some of the DEFEMON parameters use only event substitution variables &1 - &5 described above. These parameters are the following:

- LOGON (Class 0, Type 0)
- LOGOFF (Class 0, Type 1)
- Forced SLEEP started (Class 0, Type 4)
- Runnable state entered (Class 0, Type 5)
- Timebomb exploded (Class 0, Type 15)
- Forced SLEEP active (Class 1, Type 4)
- Free storage limit exceeded (Class 1, Type 6)
- Virtual system reset (Class 0, Type 27)

### DEFEMON variables &6 - &nn

This section provides additional details on the event substitution variables that are &6 and higher. The event substitution variables &1 - &5 are the same for all of the events, as described above, with only a few exceptions. Any exceptions are also included in the table below.



Table 111. DEFEMON event substitution variables &6 and higher

Event parameter or message	Variable #	Description
Failure condition detected (Class 0, Type 2)	&6	Reason, Length 1-3, Type Decimal
LOGOFF timeout started (Class 0, Type 3)	&6	Interval, Length 1 - 5, Type Decimal
Virtual machine outbound relocation started (Class 0, Type 9)	&6	Destination system, Length 1-8, Type Character
Virtual machine inbound relocation started (Class 0, Type 10)	&6	Source system, Length 1-8, Type Character
Virtual machine outbound relocation complete (Class 0, Type 11)	&6	Destination system, Length 1-8, Type Character
Virtual machine inbound relocation complete (Class 0, Type 12)	&6	Source system, Length 1-8, Type Character
Virtual machine outbound relocation terminated (Class 0, Type 13)	&6	Reason, Length 1-3, Type Decimal
Virtual machine inbound relocation terminated (Class 0, Type 14)	&6	Reason, Length 1-3, Type Decimal
Failure condition exists (Class 1, Type 2)	&6	Reason, Length 1-3, Type Decimal
LOGOFF timeout service (Class 1, Type 3)	&6	Interval remaining, Length 1-5, Type Decimal
Virtual machine outbound relocation active (Class 1, Type 9)	&6	Destination system, Length 1-8, Type Character
Virtual machine inbound relocation active (Class 1, Type 10)	&6	Source system, Length 1-8, Type Character
SSI mode change (Class 2, Type 7)	&3	SSI name, Length 1-8, Type Character
	&6	Previous SSI mode, Length 2, Type Hex
	&7	New SSI mode, Length 2, Type Hex
SSI member state change (Class 2, Type 8)	&3	SSI name, Length 1-8, Type Character
	&6	Previous SSI mode, Length 2, Type Hex
	&7	New SSI mode, Length 2, Type Hex
	&8	New SSI mode, Length 2, Type Hex
SSI mode (Class 3, Type 7)	&3	SSI name, Length 1-8, Type Character
	&6	Current <sup>®</sup> SSI mode, Length 2, Type Hex

Table 111. DEFEMON event substitution variables &6 and higher (continued)

Event parameter or message	Variable #	Description
	&7	Current SSI mode, Length 2, Type Hex
SSI member state (Class 3, Type 8)	&3	SSI name, Length 1-8, Type Character
	&6	Current SSI mode, Length 2, Type Hex
	&7	Current SSI mode, Length 2, Type Hex
	&8	Current SSI mode, Length 2, Type Hex
Device activated, LAN now operational (Class 4, Type 16)	&3	VSWITCH name, Length 1-8, Type Character
	&6	Port group name, Length 1-8, Type Character
	&7	RDEV number, Length 4, Type Hex
	&8	Port number, Length 2, Type Hex
	&9	Port status, Length 2, Type Hex
	&10	Port status reason, Length 4, Type Hex
	&11	VSWITCH status, Length 2, Type Hex
	&12	RDEV error status, Length 2, Type Hex
Additional device activated, LAN previously operational (Class 4, Type 17)	&3	VSWITCH name, Length 1-8, Type Character
	&6	Port group name, Length 1-8, Type Character
	&7	RDEV number, Length 4, Type Hex
	&8	Port number, Length 2, Type Hex
	&9	Port status, Length 2, Type Hex
	&10	Port status reason, Length 4, Type Hex
	&11	VSWITCH status, Length 2, Type Hex
	&12	RDEV error status, Length 2, Type Hex
Device deactivated, LAN remains operational (Class 4, Type 18)	&3	VSWITCH name, Length 1-8, Type Character
	&6	Port group name, Length 1-8, Type Character
	&7	RDEV number, Length 4, Type Hex
	&8	Port number, Length 2, Type Hex
	&9	Port status, Length 2, Type Hex
	&10	Port status reason, Length 4, Type Hex
	&11	VSWITCH status, Length 2, Type Hex
	&12	RDEV error status, Length 2, Type Hex
Device deactivated, LAN no longer operational (Class 4, Type 19)	&3	VSWITCH name, Length 1-8, Type Character
	&6	Port group name, Length 1-8, Type Character

Table 111. DEFEMON event substitution variables &6 and higher (continued)		
Event parameter or message	Variable #	Description
	&7	RDEV number, Length 4, Type Hex
	&8	Port number, Length 2, Type Hex
	&9	Port status, Length 2, Type Hex
	&10	Port status reason, Length 4, Type Hex
	&11	VSWITCH status, Length 2, Type Hex
	&12	RDEV error status, Length 2, Type Hex
Device activated, bridge port operational (Class 4, Type 20)	&3	VSWITCH name, Length 1-8, Type Character
	&6	Bridge port RDEV number, Length 4, Type Hex
	&7	Bridge port status, Length 2, Type Hex
	&8	Bridge port error status, Length 2, Type Hex
Device deactivated, bridge port no longer operational (Class 4, Type 21)	&3	VSWITCH name, Length 1-8, Type Character
	&6	Bridge port RDEV number, Length 4, Type Hex
	&7	Bridge port status, Length 2, Type Hex
	&8	Bridge port error status, Length 2, Type Hex
A Global virtual switch member had a connectivity problem and the local system using the same global virtual switch and shared port group took over the MAC address to continue to provide connectivity. The MAC Address assigned to Virtual Switch Name (with associated Port Group Name, PCHID Number, Port Number, and System Identifier) has been assigned to Local Virtual Switch Name on Local System Identifier. (Class 4, Type 22)		
	&3	Virtual Switch Name, Length 8, Type Character
	&6	System Identifier, Length 8, Type Character
	&7	Local Virtual Switch Name, Length 8, Type Character
	&8	Local System Identifier, Length 8, Type Character
	&9	MAC Address, Length 12, Type Hex
	&10	Port Group Name, Length 8, Type Character
	&11	PCHID Number, Length 4, Type Hex
	&12	Flags, Length 2, Type Hex

Table 111. DEFEMON event substitution variables &6 and higher (continued)

Event parameter or message	Variable #	Description
	&13	Port number, Length 1 - 3, Type Decimal
<p>A connectivity problem affecting a global virtual switch and shared port group member has been repaired or a different global virtual switch member has taken over the MAC Address. The MAC Address previously assigned to Local Virtual Switch Name and Local System Identifier is now assigned to Virtual Switch Name with associated Port Group Name, PCHID Number, Port Number, and System Identifier.</p> <p>(Class 4, Type 23)</p>		
	&3	Virtual Switch Name, Length 8, Type Character
	&6	System Identifier, Length 8, Type Character
	&7	Local Virtual Switch Name, Length 8, Type Character
	&8	Local System Identifier, Length 8, Type Character
	&9	MAC Address, Length 12, Type Hex
	&10	Port Group Name, Length 8, Type Character
	&11	PCHID Number, Length 4, Type Hex
	&12	Flags, Length 2, Type Hex
	&13	Port number, Length 1 - 3, Type Decimal
<p>The System has successfully joined the IVL Domain.</p> <p>(Class 4, Type 24)</p>	&3	Local System Identifier, Length 8, Type Character
	&6	IVL Domain, Length 2, Type Hex
	&7	Flags, Length 2, Type Hex
<p>The System has left the IVL Domain.</p> <p>(Class 4, Type 25)</p>	&3	Local System Identifier, Length 8, Type Character
	&6	IVL Domain, Length 2, Type Hex
	&7	Flags, Length 2, Type Hex
<p>An event with no defined formatting routine was received</p> <p>(Class and Type are both "Other")</p>	&3	Constant "*UNKNOWN", Length 8, Type Character
	&6	Event data, Length 0-80, Type Hex

## DEFIMON substitution variables

The DEFIMON command allows the substitution variables listed in the table below.

<i>Table 112. DEFIMON substitution variables</i>	
<b>Variable</b>	<b>Value</b>
&0	Constant "IDLE"
&1	Name of the idle monitor
&2	Name of the idle monitor's action
&3	Type of resource that is monitored: RULE, MACHINE, SCHEDULE, SPOOL, EVENT, PAGE, or IDLE.
&4	Name of the monitored resource
&5	Indicates the number of times the monitored resource was triggered within the interval specified on the OCCUR operand.

### DEFMMON substitution variables

The DEFMMON command allows the substitution variables listed in the table below.

<i>Table 113. DEFMMON substitution variables</i>	
<b>Variable</b>	<b>Value</b>
&0	Constant "MACHINE"
&1	Name of the machine monitor
&2	Name of the machine monitor's action
&3	User ID that caused the action

### DEFPMON substitution variables

The DEFPMON command allows the substitution variables listed in the table below.

<i>Table 114. DEFPMON substitution variables</i>	
<b>Variable</b>	<b>Value</b>
&C	0
&T	Concatenation of &0 &1 &2 &3 &4 &5 &6
&U	Constant _GOMPMON
&0-&99	Specifies information about the page monitor and page space. Valid values are: <ul style="list-style-type: none"> <li>• &amp;0 - Constant "PAGE"</li> <li>• &amp;1 - Name of the page space monitor</li> <li>• &amp;2 - Name of the page space monitor's action</li> <li>• &amp;3 - Condition field that is passed to the action (USAGE or INCREASE)</li> <li>• &amp;4 - Page usage percent</li> <li>• &amp;5 - Page usage percent increase for this monitor</li> <li>• &amp;6 - Interval of the page monitor</li> </ul>

### DEFSMON substitution variables

The DEFSMON command allows the substitution variables listed in the table below.

<i>Table 115. DEFSMON substitution variables</i>	
Variable	Value
&C	0
&T	The concatenation of &0 &1 &2 &3 &4 &5 &6 &7 &8
&U	The constant _GOMSMON
&0-&99	Specifies information about the spool monitor, spool usage, and spool file usage. Valid values are: <ul style="list-style-type: none"> <li>• &amp;0 - Constant SP00L</li> <li>• &amp;1 - Name of the spool monitor</li> <li>• &amp;2 - Name of the spool monitor's action</li> <li>• &amp;3 - Condition field passed to the action (USAGE or INCREASE)</li> <li>• &amp;4 - Spool usage percent</li> <li>• &amp;5 - Spool usage percent increase for this monitor</li> <li>• &amp;6 - Spool file usage percent</li> <li>• &amp;7 - Spool file usage percent increase for this monitor</li> <li>• &amp;8 - Interval of the spool monitor</li> </ul>

## DEFRULE substitution variables

The DEFRULE command allows the substitution variables listed in the table below.

<i>Table 116. DEFRULE substitution variables</i>	
Variable	Value
&C	Indicates the class of the message text. Valid classes are: <ul style="list-style-type: none"> <li>• 1 - Text came from CP MSG or CP MSGNOH.</li> <li>• 2 - Text came from CP WARNING.</li> <li>• 3 - Text came from CP console output (includes OBSERVER messages).</li> <li>• 4 - Text came from CP SMSG.</li> <li>• 5 - Text came from VM console output.</li> <li>• 6 - Text came from CP error message.</li> <li>• 7 - Text came from CP informational message.</li> <li>• 8 - Text came from CP single console image facility (includes SECUSER messages).</li> <li>• 22 - Text came from Operations Manager Remote Source Input Filter. Refer to <a href="#">“GOMRSIF”</a> on page 203 for more information.</li> <li>• 23 - Text came from Operations Manager Remote Syslog Filter. Refer to GOMRSYL and GOMRSYM in <a href="#">“Usage notes”</a> on page 53 for more information.</li> </ul>
&T	Full message text
&U	User ID of the message text originator
&0-&99	Words of the text

## DEFSCHD and RUN substitution variables

The DEFSCHD and RUN commands allow the substitution variables listed in the table below.

<i>Table 117. DEFSCHD and RUN substitution variables</i>	
<b>Variable</b>	<b>Value</b>
&C	0
&T	Null
&U	Constant _GOMSCHD
&0	Constant "SCHEDULE"
&1	Name of the schedule
&2	Name of the schedule's action

### Related reference

[Usage notes](#)





## Appendix D. Accounting and Monitoring Information

This appendix describes the accounting and monitoring records generated by Operations Manager.

### Accounting record format

Operations Manager optionally generates accounting records that contain useful information about its processing. An accounting record is generated every hour and when you shutdown GOMMAIN. To retrieve this information, you can write a program to read the accounting data and generate any reports you may need. See *IBM z/VM CP Programming Service (SC24-6001)* for more information.

To generate accounting records, you must update the CP directory for user ID OPMGRM1, and any other user ID running GOMMAIN.

Each accounting record consists of 80 characters. Each column or group of columns in the record contains specific information about Operations Manager. [Table 118 on page 193](#) describes the format of each accounting record.

Table 118. Accounting record format	
Column	Contents
01-08	User ID. (Character)
09-12	GOM is displayed to indicate that the record is being generated by Operations Manager. (Character)
13-16	The length of the Operations Manager data. (Binary)
17-24	The beginning of the accounting interval. (Clock format)
25-32	The ending of the accounting interval. (Clock format)
33-36	The count of IUCV traffic. This includes all messages, messages between consoles, and so on. (Binary)
37-40	The number of rules triggered from pattern matches. (Binary)
41-44	The number of monitored machines triggered. (Binary)
45-48	The number of schedules triggered. (Binary)
49-52	The number of spool usage monitors triggered. (Binary)
53-56	The number of space page monitors triggered. (Binary)
57-60	The number of idle monitors triggered. (Binary)
61-64	The number of event monitors triggered. (Binary)
65-68	The number of actions executed. This includes actions waiting to be executed. (Binary)
69-76	The amount of wall clock time consumed executing actions. (Clock format)
77-78	Unused. (Blank)
79-79	Accounting record identification code (C0).
80-80	Accounting record identification code (C0).

## Monitoring record format

Operations Manager optionally generates monitoring records that contain useful information about its processing. A monitoring record is generated every hour and when you shutdown GOMMAIN. To retrieve this information, you can write a program to read the accounting data and generate any reports you may need. See *IBM z/VM CP Programming Service (SC24-6001)* for more information.

To generate monitoring records, you must update the CP directory for user ID OPMGRM1, and any other user ID running GOMMAIN.

Monitoring records do not have a size limit. Each column or group of columns in the record contains specific information about Operations Manager. [Table 119 on page 194](#) describes the format of each monitoring record.

Table 119. Monitoring record format	
Column	Contents
01–08	User ID. (Character)
09–12	GOM is displayed to indicate that the record is being generated by Operations Manager. (Character)
13–16	The length of the Operations Manager data. (Binary)
17–24	The beginning of the monitor interval. (Clock format)
25–32	The ending of the monitor interval. (Clock format)
33–36	The count of IUCV traffic. This includes all messages, messages between consoles, and so on. (Binary)
37–40	The number of rules triggered from pattern matches. (Binary)
41–44	The number of monitored machines triggered. (Binary)
45–48	The number of schedules triggered. (Binary)
49–52	The number of spool usage monitors triggered. (Binary)
53–56	The number of page space monitors triggered. (Binary)
57–60	The number of idle monitors triggered. (Binary)
61–64	The number of event monitors triggered. (Binary)
65–68	The number of actions executed. This includes actions waiting to be executed. (Binary)
69–76	The amount of wall clock time consumed executing actions. (Clock format)

---

## Appendix E. Log File Management

Operations Manager writes all of its messages to a log. The log provides a chronological audit trail of Operations Manager processing. It records settings, inbound and outbound message traffic, messages from users who trigger rules, when schedules occur, machine monitors, any commands issued, and so on. The information in the log can help with problem identification and resolution.

A new log is created when Operations Manager is started or when a message is logged for a new date. Because a log is created for each new day, the log file name is *sysname yyyymmdd*, where *sysname* is the z/VM system name, *yyyy* is the year, *mm* is the month, and *dd* is the day. If a log file for a specific day already exists when the log is started, the log output will be appended to that day's log file.

Operations Manager places the log files on DASD accessed as file mode D. By default, this is the OPMGRM1 194 minidisk or the VMSYS:OPMGRM1.LOGS directory in SFS. If the D disk does not exist, no log files will be created. If the D disk location changes while Operations Manager is running, any new log files are written to the current D disk.

Before a new log file is created for a new date, Operations Manager runs GOMCLG EXEC to manage the space containing the log files. Before starting a new log file, GOMCLG removes the oldest log files on the D disk until the space of the D disk is less than 50%. If the log files are in a shared file system, Operations Manager cannot manage the log files because it cannot determine the amount of space within the shared file system to be allocated for log files. In this case, you can write your own space management routines to manage the log files within the shared space.

**Note:** It is recommended that you retain old log files for several days. You can do this by using the DEFSCHD command to automatically archive older log files in *IBM Archive Manager for z/VM* on a regular basis.

---

### Customize log file management

You can customize the log file management cleanup routine to suit your requirements. To customize the cleanup routine, you must copy and change the GOMCLG EXEC distributed with Operations Manager. The following steps provide a guideline to customize your cleanup routine:

1. Log on to the VM account your installation uses to modify Operations Manager configuration files. Do *not* use OPMGRM1 or GOMSVM 1–4.
2. VMLINK to the minidisk that contains the original GOMCLG EXEC:

```
VMLINK OPMGRM1 400 < * G RR >
```

3. VMLINK to the minidisk containing the configuration files:

```
VMLINK OPMGRM1 198 < * E MR >
```

4. Copy the original GOMCLG EXEC file:

```
COPYFILE GOMCLG EXEC G = = E
```

5. Modify the log cleanup EXEC:

```
XEDIT GOMCLG EXEC E
```

6. Release and detach the minidisk containing the configuration files.

```
VMLINK OPMGRM1 198 < DET > ( WRITE
```

7. Release and detach the minidisk containing the original GOMCLG EXEC.

```
VMLINK OPMGRM1 400 < DET >
```



---

## Appendix F. Using internet protocol and networked components

This appendix describes how to configure Operations Manager to interact with other networking software using the DEFIPCS configuration statement. This may be used to process syslog data and file input from local or remote z/VM systems, resolve DSN names, and connect instances of Operations Manager.

---

### Domain Name System (DSN) name resolution

Operations Manager may be configured to allow a DNS name instead of an IP addresses. This allows use of a DNS name in the ADDR operand of the DEFIPCS command.

To utilize DNS name resolution, the connection(s) to the DNS resolver(s) must be defined. Operations Manager supports multiple DNS resolvers, using IPV4 or IPV6 sessions and using TCP or UDP protocol. In addition, Operations Manager supports IPV4 (Type A) and IPV6 (Type AAAA) DNS resource records.

### Resolving a DSN name

To configure Operations Manager to resolve a DNS name, issue the DEFIPCS command. The following command is an example of an IP session to a DNS name resolver.

```
DEFIPCS NAME NS01 +  
APPL GOMRSNT +  
USER TCPIP +  
ADDR 173.32.88.12 +  
PORT 53 +  
PARM UDP
```

The command above starts the name resolver processing program provided by Operations Manager (APPL GOMRSNT). It then instructs the program to connect on port 53 (PORT 53) to the DNS resolver at the IPV4 address 173.32.88.12 (ADDR 173.32.88.12), which is assigned to the TCP/IP service machine named TCPIP (USER TCPIP) using UDP protocol (PARM UDP).

1. The PARM value is column specific. For more information on the available PARM values, see [“DEFIPCS” on page 52](#).
2. Use the IP address (not the DNS name) in the ADDR operand.
3. Depending on your z/VM TCP/IP configuration, you may need to authorize OPMGRM1 (and any user ID running GOMMAIN) to use the ports specified in your DEFIPCS statements. Refer to the PORT or OBEY statements in the *z/VM TCP/IP Planning and Customization* (SC24–6125) documentation for information on enabling an application to use specific ports.

### Using multiple DSN resolvers

To configure Operations Manager to use multiple DNS resolvers, a DEFIPCS command is required for each DNS resolver. The following commands provide examples of assigning IP sessions to four DNS name resolvers and an IP session which requires DNS resolution.

```
DEFIPCS NAME NS01 +  
APPL GOMRSNT +  
USER TCPIP +  
ADDR 192.168.88.12 +  
PORT 53 +  
PARM UDP
```

```
DEFIPCS NAME NS02 +  
APPL GOMRSNT +  
USER TCPIP +  
ADDR 192.168.8.10 +
```

```
PORT 53 +  
PARM TCP
```

```
DEFIPCS NAME NS03 +  
APPL GOMRSNT +  
USER TCPIP +  
ADDR 2001::ACF1:3145:2345:0003 +  
PORT 53 +  
PARM UDP
```

```
DEFIPCS NAME NS04 +  
APPL GOMRSNT +  
USER TCPIP +  
ADDR 2002::ADFB:3145:2345:0003 +  
PORT 53 +  
PARM TCP
```

```
DEFIPCS NAME SYSTEMA +  
APPL GOMRSCC +  
USER TCPIP +  
ADDR SYSA.FULLDNS.COM +  
PORT 43001
```

1. The IP session definition for SYSTEMA requires the fully qualified domain name in the ADDR operand.
2. The order of resource records requested from the DNS resolver is dependent upon the IP session address (IPV4 or IPV6).
  - If IPV4 is used, type A resource records are requested before type AAAA.
  - If IPV6 is used, type AAAA resource records are requested before type A.A response of either type A or type AAAA is accepted.

## Sharing an action processing environment

To configure an instance of Operations Manager to allow sharing of its action processing environment, issue the DEFIPCS command. The following command is an example of sharing the action processing environment of an Operations Manager instance:

```
DEFIPCS NAME LSTNENV +  
APPL GOMRSCCL +  
USER TCPIP +  
ADDR 0.0.0.0 +  
PORT 43001
```

The command above starts the listener program provided by Operations Manager (APPL GOMRSCCL) and instructs the program to listen on TCP port 43001 (PORT 43001) of any available IP address (ADDR 0.0.0.0) assigned to the TCP/IP service machine named TCPIP (USER TCPIP).

**Note:** Depending on your z/VM TCP/IP configuration, you may need to authorize OPMGRM1 (and any user ID running GOMMAIN) to use the ports specified in your DEFIPCS statements. Refer to the PORT or OBEY statements in the *z/VM TCP/IP Planning and Customization* (SC24–6125) documentation for information on enabling an application to use specific ports.

## Using processing environment of another instance

To configure an instance of Operations Manager to use the action processing environment of another instance, issue the DEFIPCS command. The following command is an example of utilizing the action processing environment of another Operations Manager instance:

```
DEFIPCS NAME SYS1 +  
APPL GOMRSCC +  
USER TCPIP +  
ADDR SYS1.FULLDNS.COM +  
PORT 43001
```

The command above starts the connector program provided by Operations Manager (APPL GOMRSCC) and instructs the program to connect on TCP port 43001 (PORT 43001) to the DNS name SYS1.FULLDNS.COM (ADDR SYS1.FULLDNS.COM) assigned to the TCP/IP service machine named TCPIP (USER TCPIP).

**Note:** Depending on your z/VM TCP/IP configuration, you may need to authorize OPMGRM1 (and any user ID running GOMMAIN) to use the ports specified in your DEFIPCS statements. Refer to the PORT or OBEY statements in the *z/VM TCP/IP Planning and Customization* (SC24–6125) documentation for information on enabling an application to use specific ports.

In addition to defining the IP sessions between Operations Manager instances, the *procsys* field of the ENV operand for an action, rule, schedule, or monitor must specify the IP session name (*ipcsid*) of the IP session connecting the remote instance. The following is an example of specifying the *procsys* field of the ENV operand:

```
RUN COMMAND 'CP QUERY USERS' ENV LVM,SYS1
```

The command above schedules the processing of the command 'CP QUERY USERS' on the local virtual machine (LVM) of the remote instance connection defined with the *ipcsid* SYS1 (SYS1).

## Sharing processing environment of multiple instances

To configure Operations Manager to share action processing environments of multiple instance, a DEFIPCS command is required for each instance. The following commands show an example of three systems. The system names are DALLAS, DETROIT, and DENVER. Each of the systems share its action processing environment, and each of the systems use the action processing environment of other instances.

### On DALLAS system:

```
DEFIPCS NAME LSTNENV APPL GOMRSCC USER TCPIP ADDR :: PORT 43001
DEFIPCS NAME DETROIT APPL GOMRSCC USER TCPIP ADDR mich-opcenter.myorg.com PORT 43001
DEFIPCS NAME DENVER APPL GOMRSCC USER TCPIP ADDR colorado-admin.myorg.com PORT 43001
```

### On DETROIT system:

```
DEFIPCS NAME LSTNENV APPL GOMRSCC USER TCPIP ADDR :: PORT 43001
DEFIPCS NAME DALLAS APPL GOMRSCC USER TCPIP ADDR texas-ops-01.myorg.com PORT 43001
DEFIPCS NAME DENVER APPL GOMRSCC USER TCPIP ADDR colorado-admin.myorg.com PORT 43001
```

### On DENVER system

```
DEFIPCS NAME LSTNENV APPL GOMRSCC USER TCPIP ADDR :: PORT 43001
DEFIPCS NAME DALLAS APPL GOMRSCC USER TCPIP ADDR texas-ops-01.myorg.com PORT 43001
DEFIPCS NAME DETROIT APPL GOMRSCC USER TCPIP ADDR mich-opcenter.myorg.com PORT 43001
```

The following is an example of an hourly schedule and a chain of actions defined on the DALLAS system:

```
DEFACTN NAME QUID1 COMMAND 'CP QUERY USERID' ENV LVM,N,*,N NEXTACTN QUID2
DEFACTN NAME QUID2 COMMAND 'CP QUERY USERID' ENV SVM,N,DETROIT,N NEXTACTN QUID3
DEFACTN NAME QUID3 COMMAND 'CP QUERY USERID' ENV LVM,N,DENVER,N
DEFSCHD NAME HHH1 WHEN 00 ACTION QUID1
```

The first action processes at the local system in the local virtual machine, the second action processes at the DETROIT system in any action processing service machine, and the third action processes at the DENVER system in the local virtual machine.

### Related concepts

[Sharing processing environment of multiple instances](#)

### Related reference

[DEFACTN](#)

[DEFSCHD](#)

[DEFIPCS](#)

## Collection and processing of syslog data

Operations Manager may be configured to accept and process syslog data from any machine that supports forwarding syslog data. When Operations Manager receives syslog data, the data is processed in the same manner as SECUSER or OBSERVER data from a monitored user console; the data is filtered through the defined rules and triggers actions if any matching criteria is found. The collected syslog data may be viewed using the VIEWCON command.

To configure Linux, or other syslog capable machines, to forward syslog data to another system, please consult your Linux documentation. The data must be sent to the IP address of the system running Operations Manager.

To configure Operations Manager to process syslog data, issue the DEFIPCS command. DEFIPCS supports receiving syslog data using UDP and TCP protocol.

- The processing program, GOMRSL, supports receiving syslog data using UDP protocol.
- The processing program, GOMRSL, supports receiving syslog data using TCP protocol. The PTF for APAR PI74064 is required to use this function

To process syslog data using TCP protocol, the syslog data must be formatted according to the “octet counting” framing outlined in RFC 6587. To support receiving syslog data using UDP and TCP protocol, use separate DEFIPCS commands. The following command is an example for UDP protocol:

```
DEFIPCS NAME RMTSYLGD +  
APPL GOMRSL +  
USER TCPIP +  
ADDR 0.0.0.0 +  
PORT 514 +  
PARM 'SYSLOG 03330417UTF8'
```

The command above starts the syslog processing program, GOMRSL, provided by Operations Manager (APPL GOMRSL) and instructs the program to listen on UDP port 514 (PORT 514) of any available IP address (ADDR 0.0.0.0) assigned to the TCP/IP service machine named TCPIP (USER TCPIP). The PARM field establishes the following information:

- the user ID the syslog data is collected under (SYSLOG)
- the ASCII to EBCDIC translation table used (03330417)
- the ASCII encoding scheme (UTF8)

The following command is an example for TCP protocol:

```
DEFIPCS NAME RMTSYLOG +  
APPL GOMRSL +  
USER TCPIP +  
ADDR 0.0.0.0 +  
PORT 610 +  
PARM 'SYSLOG 03330417UTF8'
```

The command above starts the syslog processing program, GOMRSL, provided by Operations Manager (APPL GOMRSL) and instructs the program to listen on TCP port 610 (PORT 610) of any available IP address (ADDR 0.0.0.0) assigned to the TCP/IP service machine named TCPIP (USER TCPIP). The PARM field establishes the following information:

- the user ID the syslog data is collected under (SYSLOG)
- the ASCII to EBCDIC translation table used (03330417)
- the ASCII encoding scheme (UTF8)

### Notes:

- The PARM value is column specific. The DEFIPCS command provides more information on the available PARM values.
- Depending on your z/VM TCP/IP configuration, you may need to authorize OPMGRM1 (and any user ID running GOMMAIN) to use the ports specified in your DEFIPCS statements. Refer to the PORT or OBEY



statements in the *z/VM TCP/IP Planning and Customization (SC24–6125)* documentation for information on enabling an application to use specific ports.

To view the collected syslog data, issue the VIEWCON command:

```
GOMCMD OPMGRM1 VIEWCON USER SYSLOG
```

where OPMGRM1 is the user ID of the Operations Manager server running GOMMAIN.

## Translation tables

The syslog data must be converted from ASCII to EBCDIC. Three sample ASCII to EBCDIC translation tables are provided on the sample installation disk and may be copied to the OPMGRM1 198 minidisk or any other disk accessed by user ID OPMGRM1, or any other user ID running GOMMAIN. Operations Manager allows the use of any input and output code page.

Table 120 on page 201 lists the sample tables included in the Operations Manager samples disk 5697J10F 2C2.

Table 120. Sample translation table files	
Table file name	Description
03330025 SAMPTABL	Translation from ASCII code page 00819 to EBCDIC code page 00037
03330417 SAMPTABL	Translation from ASCII code page 00819 to EBCDIC code page 01047
0333047C SAMPTABL	Translation from ASCII code page 00819 to EBCDIC code page 01148
00000000 SAMPTABL	No translation is performed. This sample table can be used when data is expected to be in EBCDIC when received.

IBM also provides many translation tables at <http://www.ibm.com/software/globalization/cdra/index.jsp>. Refer to Appendix J for a link to downloadable translation tables. When the download is complete, the file must have fixed records and a blocksize of 256 for Operations Manager to accept the translation table.

The syslog data must be encoded in ISO8 or UTF8 format. If the syslog data is encoded in UTF8, the input is limited to plane0 and plane1 of Unicode.

## Copying the sample Operations Manager tables

To copy the sample translation tables provided with Operations Manager, follow the steps below.

1. Log on to the VM account your installation uses to modify Operations Manager configuration files. Do *not* use OPMGRM1 or OPMGRS1–4.
2. VMLINK to the minidisk that contains the samples:

```
VMLINK 5697J10F 2C2 < * I RR >
```

3. VMLINK to the minidisk that contains the configuration files:

```
VMLINK OPMGRM1 198 < * E MR >
```

4. Copy the translation tables:

```
COPYFILE * SAMPTABL I = TABLE E
```

5. Release and detach the minidisk that contains the samples:

```
VMLINK OPMGRM1 2C2 < DET >
```

6. Release and detach the minidisk that contains the samples:

```
VMLINK OPMGRM1 198 < DET > ( WRITE
```

## Collection and processing of remote file input

Operations Manager may be configured to accept and process the content of a file sent from a CMS guest on a local or remote z/VM system. When Operations Manager receives the file content, the data is processed in the same manner as SECUSER or OBSERVER data from a monitored user console; the file content is filtered through the defined rules and triggers actions if any matching criteria is found. The collected file content may be viewed using the **VIEWCON** command.

To configure Operations Manager to process file content data, issue the **DEFIPCS** command. The following command is an example:

```
DEFIPCS NAME RMTFILES +  
APPL GOMRSIF +  
USER TCPIP +  
ADDR 0.0.0.0 +  
PORT 63000
```

The command above starts the file content processing program provided by Operations Manager (APPL GOMRSIF) and instructs the program to listen on TCP port 63000 (PORT 63000) of any available IP address (ADDR 0.0.0.0) using the TCP/IP service machine named TCPIP (USER TCPIP).

### Notes:

1. The port specified on the PORT operand of the **DEFIPCS** command must match the port specified on the **GOMRSIF** command.
2. The port specified on the PORT operand of the **DEFIPCS** command must be enabled in TCP/IP for the TCP protocol. The **GOMRSIF** command only supports the TCP protocol.

## Send and view file content

To send file content from a CMS guest, issue the GOMRSIF EXEC:

```
GOMRSIF TESTCONS DATA A 192.168.100.7 63000 TERRY
```

The command above transfers the content of TESTCONS DATA A to Operations Manager at the IP address 192.168.100.7 on port 63000 and the data is collected under user name TERRY.

To view the collected file content, issue the VIEWCON command:

```
GOMCMD OPMGRM1 VIEWCON USER TERRY
```

Where OPMGRM1 is the user ID of the Operations Manager server running GOMMAIN.

## Send and view multiple files

To send multiple files to Operations Manager and view them all in one VIEWCON session, follow the steps below.

1. Issue a DEFIPCS statement in Operations Manager. For an example, refer to [“Collection and processing of remote file input”](#) on page 202.
2. From a VM user ID that has access to the files, issue the necessary GOMRSIF commands. The port specified in the DEFIPCS statement must match the port specified in the GOMRSIF command. The following example sends the content of TEST DATA A and SAMPLE DATA A to Operations Manager and specifies that the data is collected under the user name TERRY:

```
GOMRSIF TEST DATA A 192.168.100.200 63000 TERRY  
GOMRSIF SAMPLE DATA A 192.168.100.200 63000 TERRY
```

3. This allows you to view all of the remote file input in one VIEWCON session. For this example, the following command is issued to display the data:

```
GOMCMD OPMGRM1 VIEWCON USER TERRY
```

To send multiple files to Operations Manager and view them in separate VIEWCON sessions, follow the steps below.

**Note:** Only one DEFIPCS statement is required.

1. Issue a DEFIPCS statement in Operations Manager. For an example, refer to [“Collection and processing of remote file input”](#) on page 202.
2. From a VM user ID that has access to the files, issue the necessary GOMRSIF commands, specifying different user names. The following example sends the content of TEST DATA A to Operations Manager and specifies that the data is collected under the user name TERRY, then sends the content of SAMPLE DATA A and specifies that the data is collected under the user name JOE.

```
GOMRSIF TEST DATA A 192.168.100.200 63000 TERRY
GOMRSIF SAMPLE DATA A 192.168.100.200 63000 JOE
```

3. This allows you to view the remote file input in separate VIEWCON sessions. For this example, the following commands are issued to display the data:

```
GOMCMD OPMGRM1 VIEWCON USER TERRY
GOMCMD OPMGRM1 VIEWCON USER JOE
```

The following section describes the GOMRSIF EXEC in more detail.

## GOMRSIF

---

➡ GOMRSIF — *filename* — *filetype* — *filemode* — *ipaddr* — *ipport* — *username* ➡

Figure 134. GOMRSIF command syntax

## Authorization

No specific Operations Manager authority is required for the GOMRSIF command. This command may be run on any system where the EXEC is installed; Operations Manager does not need to be installed to issue the GOMRSIF command.

## Purpose

Issue the GOMRSIF command to send a file to Operations Manager and collect the data under a specified user name.

## Operands

### ***filename***

Specifies the file name.

### ***filetype***

Specifies the file type.

### ***filemode***

Specifies the file mode.

### ***ipaddr***

Specifies the target VM IP address to send the data. Format is xxx.xxx.xxx.xxx. Leading zeros are not required. This must be the IP address of the system running Operations Manager.

### ***ipport***

Specifies the target VM IP port to send the data. Format is nnnnn. Leading zeros are not required. This must match the port number specified on the corresponding DEFIPCS statement.

### ***username***

Specifies the user name the data is collected under.

## Usage notes

None

## Return codes

[Table 121 on page 204](#) lists the return codes for the GOMRSIF command.

Table 121. GOMRSIF return codes	
Return code	Description
0	Successful
4	No data in file
8	Error reading file
12	File not found or TCP/IP error
16	Command parameter missing or parameter error

---

# Appendix G. Testing Operations Manager installation and configuration

This appendix describes the steps required to test various components of Operations Manager's installation and configuration.

If any of the tests described in this appendix fail, you must update the Operations Manager configuration file.

---

## Checking the status of OPMGRM1

To verify that the Operations Manager server (OPMGRM1) has been successfully started and is able to process requests, follow the steps below.

1. Log on to an authorized user ID. This user ID must be authorized to issue the Operations Manager STATUS command. This can be done by giving the user configuration authority, control authority, or authorizing the user to the STATUS command.
2. Issue the following command to instruct Operations Manager to output status information:  

```
GOMCMD OPMGRM1 STATUS
```
3. Several messages from OPMGRM1 appear indicating the server's status.

---

## Verifying consoles are monitored and viewable

To verify that the console of a user ID is being monitored by Operations Manager and that authorized users can view and interact with the console follow the steps below.

1. Log on to an authorized user ID. This user ID must be authorized to view a monitored console using the VIEWCON command.
2. Issue the following command to view another user's console:

```
VIEWCON userid
```

Where *userid* is a monitored console that has or is generating console messages. *userid* must have SECUSER or OBSERVER set to OPMGRM1. In addition, if you are using SECUSER, *userid* must be disconnected.

3. A full-screen view of the console of *userid* appears. Use PF07 and PF08 to scroll up and down in the console, PF10 and PF11 to scroll left and right, and PF01 to toggle autoscrolling.

**Note:** Any commands entered on the VIEWCON command line are issued on *userid*.

4. Press PF03 to exit the VIEWCON session.
5. Issue the following command to view the Operations Manager log:

```
VIEWLOG
```

6. A full-screen view of the console of the Operations Manager server (OPMGRM1) appears. This is also known as the Operations Manager log. The log includes all consoles being monitored by Operations Manager as well as messages concerning rules, schedules, monitors, and actions. You can also use the PF keys to scroll up, down, left, right, and toggle autoscrolling.
7. Press PF03 to exit the VIEWLOG session.

## Creating a schedule and corresponding action

---

To create a schedule and corresponding action to send a message to your user ID five minutes from now, follow the steps below. In this example, you are defining the schedule and action dynamically. Typically, schedules and actions are defined in the configuration file that is loaded when Operations Manager starts.

1. Log on to an authorized user ID. This user ID must be authorized to define schedules and define actions. This can be done by giving the user ID configuration authority or authorization to the DEFSCHD and DEFACTN commands.
2. To define an action to send you a message, issue the following command:

```
GOMCMD OPMGRM1 DEFACTN NAME TSTACTN1 +  
COMMAND 'CP MSG userid Test Operations Mgr' +  
ENV LVM
```

Replace *userid* with your current user ID.

3. To define a schedule that triggers in five minutes from now, issue the following command :

```
GOMCMD OPMGRM1 DEFSCHD NAME TESTSCHD +  
ACTION TSTACTN1 +  
WHEN NOW+00:05
```

4. In five minutes the following message from OPMGRM1 appears:

Test Operations Mgr

## Verifying spool files are viewable

---

To verify that authorized users can view and purge spool files, follow the steps below.

1. Log on to an authorized user ID. This user ID must be authorized to view spool files using the VIEWSPL command.
2. Issue the following command to view the list of spool files on the system:

```
VIEWSPL
```

3. A full-screen view of the spool files on the system appears. Use PF07 and PF08 to scroll up and down in the list. Use PF10 and PF11 to scroll left and right.
  - To sort the list, place the cursor on a column heading and use PF05 or PF06 to sort ascending or descending.
  - To view the contents of a file, place the cursor on the correct row, and use PF02. When viewing the contents of a file, use PF03 to exit the file and return to the list view. Files noted with an asterisk (\*) are still open and not viewable.
  - To purge a file, place the cursor in the command area next to the file and type PURGE. Files noted with an asterisk (\*) are still open and can not be purged.
4. Press PF03 to exit the VIEWSPL session.

---

## Appendix H. Application and settings interference

Certain applications and settings can interfere with Operations Manager. The use of any application or setting that interferes with Operations Manager is strongly discouraged in any virtual machine while any Operations Manager component is active.

### Applications

Applications that do any of the following can interfere with Operations Manager:

- Manipulate the system clock comparator directly. The applications should use STIMER or STIMERM macros to manipulate the system clock comparator and system clock comparator interrupts.
- Utilize the \*MSG or \*MSGALL CP IUCV system services. CP system services restrict the number of IUCV connections to these services to one connection. Operations Manager requires the connection to function correctly.
- Modify the settings for CPCONIO, VMCONIO, MSG, IMMSG, WNG, EMSG, or SMSG. Operations Manager establishes the required settings at initialization, and does not expect the settings to change.

The WAKEUP program is known to violate one or more of the above rules and interfere with Operations Manager.

The following user IDs run Operations Manager components and should not run an application that will interfere with it:

- OPMGRM1 and any other user ID running GOMMAIN
- Any action processing server (OPMGRS1 through OPMGRS4) and any other user ID running GOMSVM

### Settings

Operations Manager utilizes the \*MSG system service. The output messages generated from action processing are captured by the \*MSG system service.

The following settings are known to interfere with Operations Manager's ability to capture the output messages generated from action processing using the \*MSG system service.

- Setting OBSERVER to any value other than OFF
- Setting SECUSER to any value other than OFF

The following user IDs run Operations Manager components and should not use settings that will interfere with it:

- OPMGRM1 and any other user ID running GOMMAIN
- Any action processing server (OPMGRS1 through OPMGRS4) and any other user ID running GOMSVM





---

## Appendix I. GOMGLBL facility

GOMGLBL is an optional facility intended to help share information between REXX EXECs that run as Operations Manager actions that execute on different Operations Manager servers. The servers can include the main Operations Manager server (OPMGRM1 by default) and any of the action processing servers (OPMGRSn by default).

Although GOMGLBL is intended to be used with Operations Manager, it is not required to run on or use Operations Manager and does not need to be configured unless you intend to use it. GOMGLBL is somewhat similar to the CMS GLOBALV facility, which allows you to define names, assign values to those names, and retrieve the values at a later time. Whereas CMS GLOBALV is restricted to use within a single CMS guest, GOMGLBL provides a similar function across several CMS guests.

GOMGLBL may not be suitable for every environment or you may prefer other techniques for sharing data between Operations Manager servers (such as access to a Shared File System file).

GOMGLBL requires a writable shared segment, named GOMSEG, which all of the GOM workers sharing data must be able to access. The size of the segment depends on how much data you want to share; each megabyte in the shared segment can contain more than 2000 name/value pairs.

Similar to CMS GLOBALV, the names and the values assigned to them can be up to 255 characters in length. The names must be valid as REXX variable names (no embedded spaces, and so on), but the value can contain any character.

Any user ID that will be issuing the GOMGLBL command must have authority to issue a CP diagnostic code `DIAG x'0064'`. By default, this authority is provided in the NAMESEG GOMSEG option in the user's CP directory entry.

---

### Preparing GOMGLBL

The GOMSEGDF EXEC is provided to help in defining the GOMSEG writable shared segment. As provided, GOMSEGDF will define a 2MB segment at location 0x6000000 (96MB), which will provide room for 4095 shared variables. The location of the shared segment must be beyond the defined storage of any Operations Manager server (or other CMS guest) that will use it.

As shipped, the GOMSEGDF EXEC defines GOMSEG as shared, writable, and restricted (RSTD), so that only guests which have a NAMESAVE statement for GOMSEG in their directory entry can use it.

1. Customize GOMSEGDF EXEC with the appropriate values for your site:

**Note:** If you intend to skip a large amount between the end of the defined storage for the Operations Manager servers and the start of the shared segment, review the usage notes about the effects on real storage in the DEFSEG section in the *CP Commands and Utilities Reference* for your level of z/VM.

If you require more than 4095 shared variables, each additional megabyte provides for 2048 more. For example, available shared variables = (shared segment size in MB\*2048) –1.

- a) From user ID 5697J10F, type:

```
'ACC 400 M'
'XEDIT GOMSEGDF EXEC M'
```

- b) Find the following statement:

```
Address 'COMMAND' 'CP DEFSEG GOMSEG 6000-61FF SW RSTD'
```

Change any of the following items as needed:

- Change the address range of the segment location if you want the segment larger than 2MB or located someplace other than 96MB.
- Remove the RSTD operand if you do not want the segment to be restricted.

- c) File your changes.
2. Run GOMSEGDF EXEC from a user ID with the necessary privilege class to issue the CP DEFSEG command. By default, this is class E. For example, issue the following command:

```
GOMSEGDF DEFINESEG
```

For more information on privilege classes, refer to [Chapter 2, “Configuring Operations Manager,”](#) on page 5.

3. Run the GOMSEGIN program (GOMSEGIN MODULE) to initialize the segment.
4. Ensure that all user IDs that will use GOMGLBL have the NAMESAVE GOMSEG statement in their directory entry.  
GOMGLBL is now ready to use.
5. Log off and restart any user IDs that were updated in step 4.

## GOMGLBL

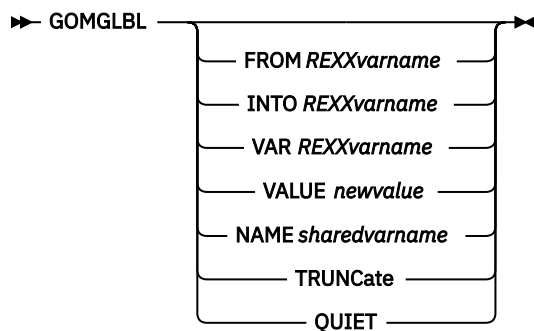


Figure 135. GOMGLBL command syntax

## Authorization

The user issuing this command must be authorized to access the GOMSEG DCSS via a NAMESAVE directory statement.

## Purpose

Issue the GOMGLBL command to set or retrieve the value of a shared variable.

## Operands

### **FROM REXXvarname**

Sets the value of the shared variable to the contents of the REXX variable *REXXvarname*. REXX must be active.

### **INTO REXvarname**

Place the value of the shared variable into the REXX variable *REXXvarname*. REXX must be active.

### **VAR REXXvarname**

Indicates the name of the shared variable that is to be obtained from the REXX variable *REXXvarname*. REXX must be active.

### **VALUE newvalue**

Sets the value of the shared variable to *newvalue*.

### **NAME sharedvarname**

The *sharedvarname* is the name of the shared variable whose value is to be set and/or retrieved.

**TRUNCate**

Specifies that if the value obtained for the FROM operand exceeds 255 bytes, truncate the value to 255 bytes and perform the set operation anyway.

**QUIET**

Does not write error messages to the console.

## Usage notes

1. Shared variable names and values are limited to 255 bytes.
2. Variables may be retrieved and set in the same operations. For example, the old value, if any, will be retrieved, and then the new value set.
3. If you attempt to retrieve a shared variable which has never been set, a null value will be returned.
4. If a REXX variable is specified on the FROM operand whose contents exceed 255 bytes in length, the operation will terminate with an error message unless the TRUNCate operand is also specified.
5. The VALUE operand may be used to provide data to be stored into a shared variable, but data can only be retrieved into a REXX variable. Data specified on the VALUE operand cannot contain embedded blanks. To store data containing embedded blanks, the FROM operand must be set.
6. The values of shared variables are not saved across IPLs of the z/VM system, or when all machines using the DCSS log off.
7. You must not use stem variables whose names contain lower-case characters with GOMGLBL.

## Restrictions

Only user IDs permitted to access the GOMSEG DCSS will be able to use GOMGLBL.

## Return codes

Table 122 on page 211 lists the return codes for the GOMGLBL command.

Table 122. GOMGLBL return codes	
Return code	Description
8	No parameters supplied.
12	A required operand is missing.
	A required parameter is missing.
	Conflicting options selected.
	Duplicate options entered.
	Operand exceeds maximum length.
	A specified REXX variable contains no data. The REXX variable specified for VAR is set to null.

Table 122. GOMGLBL return codes (continued)

Return code	Description
16	GOMGLBL requires the extended parameter list. GOMGLBL can only be called from environments, such as CMS or REXX, that support the CMS extended parameter list.
	CSL services are not available.
	VARNAME, FROM, and INTO are specified but not running under REXX. The REXX environment must be active for GOMGLBL to access REXX variables.
	GOMSEG DCSS is not initialized or is at the wrong level.
	The user is not authorized to access the GOMSEG DCSS. The GOMSEG DCSS by default is restricted; users must have a NAMESAVE GOMSEG statement in their directory entry.
	DCSS GOMSEG is not defined.
	CC=3 from diag 0x64 (FINDSEG).
	Error code <rc> (hex) from FINDSEG.
	Error code <rc> (hex) from LOADSHR.
104	Cannot retrieve value-spin.  The value of the variable specified could not be retrieved, probably due to concurrent activity from other users. GOMGLBL retries the operation several times before issuing this message, but you may also reissue the command if desired.
108	Error code <rc> (hex) from DMSCSR.  The value of the variable specified was retrieved, but could not be stored into a REXX variable for the caller. However, if a new value was specified, it was successfully stored.
112	Cannot set new value-spin.  The value of the variable specified could not be set, probably due to concurrent activity from other users. GOMGLBL retries the operation several times before issuing this message, but you may also reissue the command if desired.
116	Indicates that while the specified variable exists, and was retrieved, it could not be placed in a REXX variable for the caller, nor could the new value be stored. One or both of the messages associated with return codes 108 and 112 may be issued.
120	Out of room in the GOMSEG saved segment.
124	Data exceeds 255 bytes and TRUNC was not specified.  The FROM operand was used to direct GOMGLBL to obtain the value to which to set a shared variable, but the contents of that REXX variable were more than 255 bytes in length, and the TRUNC operand was not specified. The maximum length of the value of a shared variable is 255 bytes. If you want GOMGLBL to truncate strings when necessary, use the TRUNCate operand.
128	VAR variable exceeds 255 bytes.  The VAR operand was used to direct GOMGLBL to obtain the name of the shared variable to set and/or retrieve from a REXX variable, but the contents of that REXX variable were more than 255 bytes in length. The maximum length of a shared variable name is 255 bytes.

---

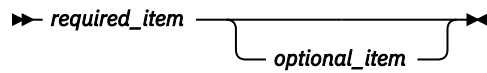
## Appendix J. How to read syntax diagrams

The following rules apply to the syntax diagrams that are used in this information:

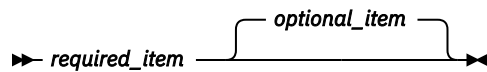
- Read the syntax diagrams from left to right, from top to bottom, following the path of the line. The following conventions are used:
  - The >>--- symbol indicates the beginning of a syntax diagram.
  - The ---> symbol indicates that the syntax diagram is continued on the next line.
  - The >--- symbol indicates that a syntax diagram is continued from the previous line.
  - The --->< symbol indicates the end of a syntax diagram.
- Required items appear on the horizontal line (the main path).

➤➤ *required\_item* ➤➤

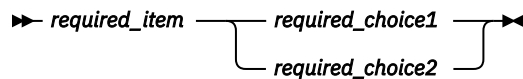
- Optional items appear below the main path.



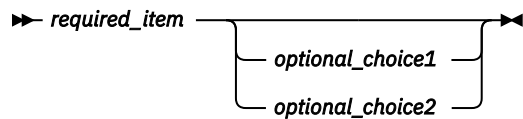
If an optional item appears above the main path, that item has no effect on the execution of the syntax element and is used only for readability.



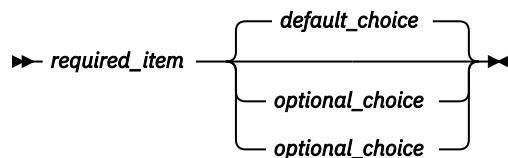
- If you can choose from two or more items, they appear vertically, in a stack.  
If you *must* choose one of the items, one item of the stack appears on the main path.



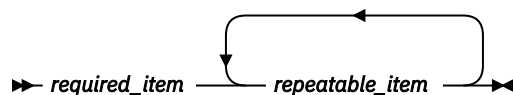
If choosing one of the items is optional, the entire stack appears below the main path.



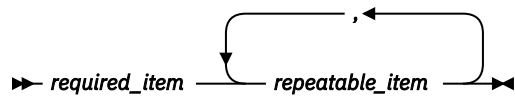
If one of the items is the default, it appears above the main path, and the remaining choices are shown below.



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



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



A repeat arrow above a stack indicates that you can repeat the items in the stack.

- Keywords, and their minimum abbreviations if applicable, appear in uppercase. They must be spelled exactly as shown. Variables appear in all lowercase italic letters (for example, *column-name*). They represent user-supplied names or values.
- Separate keywords and parameters by at least one space if no intervening punctuation is shown in the diagram.
- Enter punctuation marks, parentheses, arithmetic operators, and other symbols exactly as shown in the diagram.
- Footnotes are shown by a number in parentheses; for example, (1).

## Appendix K. Messages and Codes

This appendix lists the messages generated by Operations Manager. A description of each message and suggestions on how you can resolve the problem are provided, where possible.

Many of the messages issued by Operations Manager include a return code. For a description of these return codes, refer to the following IBM publications:

- *z/VM CP Commands and Utilities Reference* (SC24-6175)
- *z/VM CP Programming Services* (SC24-6084)
- *z/VM CP Messages and Codes* (GC24-6177)
- *z/VM CP Macros and Functions Reference* (SC24-6204)
- *z/VM CMS and REXX/VM Messages and Codes* (GC24-6118)
- *z/VM RACF Security Server Messages and Codes* (GC24-6217)
- *z/OS MVS Programming: Assembler Services Reference* (SA23-1372)

You can find all of these publications on the IBM Knowledge Center:

<https://www.ibm.com/support/knowledgecenter/>

All of the messages generated by Operations Manager have a severity code as the last character of the message ID. Severity codes are described below in [Table 123 on page 215](#).

Table 123. Severity codes	
Code	Description
I	Information only. No user action is required.
L	Log message. No user action is required.
W	Warning message. Results might not be as expected. Read the "User Response" text to determine the appropriate course of action.
E	Error. Some errors might be user-correctable, read the "User Response" text to determine the appropriate course of action.
S	Severe message. Read the "User Response" text to determine the appropriate course of action.

**GOM0001I**      **STARTED V1R6 BUILT yyyy/mm/dd**  
**hh:mm**

### Explanation

An Operations Manager component has started.

### System action

The program continues.

### User response

No action is required.

### Explanation

An attempt to establish a system timer failed. The STIMERM SET system call returned code *nn*.

### System action

The Operations Manager command terminates.

### User response

See STIMERM in the *z/OS MVS Programming: Assembler Services Reference* for a list of codes and descriptions.

**GOM0002E**      **STIMERM SET FAILED nn**

**GOM0003E**      **REQUIRED PARAMETERS NOT**  
**PRESENT**

## Explanation

Information required for startup was missing.

## System action

The Operations Manager command terminates.

### User response:

Provide the required parameters and retry the command.

---

**GOM0004I**      **EP= xxxxxxxx**

## Explanation

Records the entry point of startup, where xxxxxxxx is the actual entry point.

## System action

The program continues.

## User response

No action is required.

---

**GOM0005E**      **HNDEXT SET FAILED nn**

## Explanation

An attempt to establish an external interrupt processing routine failed. The HNDEXT SET system call returned code *nn*.

## System action

The Operations Manager command terminates.

## User response

See HNDEXT in the *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

**GOM0006W**      **ACCOUNTING INFORMATION NOT GENERATED cc**

## Explanation

An attempt to generate an accounting information record failed. The condition code from the operation is indicated in *cc*.

## System action

The program continues.

## User response

See DIAGNOSE X'4c' in the *z/VM CP Programming Services* documentation for a list of codes and descriptions.

---

**GOM0007W**      **MONITOR SAMPLE INTERVAL  
APPLDATA NOT GENERATED cc rc**

## Explanation

An attempt to generate a MONITOR SAMPLE INTERVAL record failed. The condition code and return code are indicated by *cc* and *rc*.

## System action

The program continues.

## User response

Add OPTION APPLMON to the CP directory so Operations Manager can generate MONITOR SAMPLE INTERVAL records. See DIAGNOSE x'DC' in the *z/VM CP Programming Services* documentation for a list of codes and descriptions.

---

**GOM0008E**      **COMMAND TO userid NOT  
ACCEPTED**

### Explanation:

The VM *userid* did not process a command due to version/service level incompatibility between components.

### System action:

The Operations Manager command terminates.

### User response:

Ensure the version/service level of GOMMAIN, GOMCMD, and GOMSVM are compatible. V1R6 and V1R5 components are not compatible.

---

**GOM0009E**      **SUBCOM PROCESSING LIMITED  
TO userid**

### Explanation:

When GOMCMD is processed from an ACTION, the *userid* is restricted to the *userid* running GOMMAIN which initiated the ACTION.

### System action:

The Operations Manager command terminates.

### User response:

Ensure that the *userid* specified on the GOMCMD command is correct.

---

**GOM0010E**      **GOM ENVIRONMENT ALREADY  
ACTIVATED**

### Explanation:



When an Operations Manager command is started, it attempts to establish a GOM environment. The Operations Manager command detected the GOM environment is currently active from another Operations Manager command.

**System action:**

The Operations Manager command terminates.

**User response:**

Issue the Operations Manager command on another user ID or terminate the previous Operations Manager command.

---

**GOM0011E      GOM ENVIRONMENT NOT ACTIVATED**

**Explanation:**

When an Operations Manager command is started, it attempts to establish a GOM environment. The Operations Manager command was unable to establish the GOM environment.

**System action:**

The Operations Manager command terminates.

**User response:**

Determine why the GOM environment cannot be activated.

---

**GOM0012E      EXTENDED PARAMETER LIST NOT AVAILABLE**

**Explanation**

When an Operations Manager command is started, it attempts to obtain the extended parameter list. The Operations Manager command was unable to obtain the extended parameter list.

**System action**

The Operations Manager command terminates.

**User response:**

Determine why the Operations Manager command is unable to obtain an extended parameter list.

---

**GOM0013E      NOTE: PER GOM-464, this message is no longer needed. MACHINE ARCHITECTURE NOT SUPPORTED**

**Explanation:**

When an Operations Manager command is started, it verifies the current virtual machine architecture is supported.

**System action:**

The Operations Manager command terminates.

**User response:**

The GOMMAIN command requires virtual machine architecture XC. The GOMCMD, GOMSVM, and

GOMTDM commands require virtual machine architecture ESA or XC. See *z/VM CP Commands and Utilities Reference* (SC24-6175) for additional information about the SET MACHINE command.

---

**GOM0014E      COMPONENT SOFTWARE LEVEL NOT SUPPORTED**

**Explanation:**

The GOMCMD component is not compatible with the GOMMAIN or GOMSVM component already active.

**System action:**

The Operations Manager command terminates.

**User response:**

Ensure the version/service level of GOMCMD is compatible with the version/service level of GOMMAIN and GOMSVM.

---

**GOM0099S      UNRECOVERABLE ERROR ENCOUNTERED**  
**PSW=xxxxxxxxxxxxxxxx cccccccc**  
**rrrrrrr EP=xxxxxxxx**  
**MCB=xxxxxxxx**

**Explanation**

An unrecoverable error condition was encountered.

**System action**

The Operations Manager command terminates.

**User response:**

Contact IBM Technical Support for assistance. See *z/VM CMS and REXX/VM Messages and Codes* for a list of code descriptors. The system code and reason code are represented by cccccccc and rrrrrrr.

---

**GOM0100I      ENDED**

**Explanation**

An Operations Manager component has ended.

**System action**

The Operations Manager command terminates.

**User response**

No action is required.

---

**GOM0101E      ABNEXIT SET FAILED, RC/RSN=nnnnnnnn**

**Explanation**

An attempt to establish an error recovery routine failed. The ABNEXIT SET system call returned code nnnnnnnn.

## System action

The Operations Manager command terminates.

## User response

See ABNEXIT in the *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

<b>GOM0102E</b>	<b>ABNEXIT CLR FAILED, RC/ RSN=nnnnnnnn</b>
-----------------	---

---

## Explanation

An attempt to terminate an error recovery routine failed. The ABNEXIT CLR system call returned code *nnnnnnnn*.

## System action

The Operations Manager command terminates.

## User response

See ABNEXIT in the *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

<b>GOM0111L</b>	<b>COMMAND <i>cmd</i> RC=<i>rc</i> (cmd response)</b>
-----------------	---

---

## Explanation

The CP command *cmd* returned a non-zero return code. The return code and command output are listed.

## System action

The program continues.

## User response

No action is required.

---

<b>GOM0112L</b>	<b>COMMAND <i>cmd</i> OUTPUT TRUNCATED, AVAIL=<i>avail</i> USED=<i>used</i> TRUNC=<i>trunc</i></b>
-----------------	--

---

## Explanation

The CP command *cmd* response was truncated. The number of bytes available, used, and truncated are listed.

## System action

The program continues.

## User response

No action is required.

---

<b>GOM0121I</b>	<b>MCB IS LOCATED AT <i>xxxxxxx</i></b>
-----------------	---

---

## Explanation

Records the address, represented by *xxxxxxx*, of the MCB for this component of Operations Manager.

## System action

The program continues.

## User response

No action is required.

---

<b>GOM0122I</b>	<b>VMSIZE=<i>xxxxxxx</i> ASSIZE=<i>xxxxxxx</i> ASIT=<i>xxxxxxxxxxxxxxxx</i></b>
-----------------	---

---

## Explanation

Records the virtual machine size in bytes, address space size in pages, and the address space token for this component of Operations Manager.

## System action

The program continues.

## User response

No action is required.

---

<b>GOM0123I</b>	<b>PRIVILEGES:</b>
-----------------	--------------------

---

## Explanation

This message displays in combination with message 0124I. The result is a message that records the CP PRIV classes available for this component of Operations Manager.

## System action

The program continues.

## User response

No action is required.

---

<b>GOM0124I</b>	<b>(list of privileges)</b>
-----------------	-----------------------------

---

## Explanation

This message displays in combination with 0123I. The result is a message that records the CP PRIV classes available for this component of Operations Manager.

## System action

The program continues.

## User response

No action is required.

---

**GOM0125I        SETTINGS:**

## Explanation

This message displays in combination with message 0126I. The result is a message that records the CP SETTINGS in effect at the beginning of this component of Operations Manager.

## System action

The program continues.

## User response

No action is required.

---

**GOM0126I        (list of settings)**

## Explanation

This message displays in combination with 0125I. The result is a message that records the CP SETTINGS in effect at the beginning of this component of Operations Manager.

## System action

The program continues.

## User response

No action is required.

---

**GOM0129I        STARTUP COMMAND *cmd***

## Explanation

Records the startup command and parameters entered to begin this component of Operations Manager.

## System action

The program continues.

## User response

No action is required.

---

**GOM0130W        DIAGNOSE CODE *hhhh* NOT  
AVAILABLE**

## Explanation

The privileged diagnose code *hhhh* is not available to Operations Manager. You may be unable to use one or more features of Operations Manager.

## System action

The program continues. One or more features of Operations Manager are unavailable.

## User response:

Change the privilege class to allow the diagnose code listed if the particular feature of Operations Manager is required. See “Step 1. Update OPMGRM1” on page 5 for a list of privilege classes for diagnose codes used by Operations Manager.

---

**GOM0151L        RACROUTE RETURNED *nn* CODE *cc*  
REASON *rr* (*esmprof*)**

## Explanation:

A request for security authorization returned an unexpected value. The RACROUTE AUTH system call returned an unexpected value. An attempt to access the external security manager profile *esmprof* returned SAF code *nn*, return code *cc*, and reason code *rr*. See the appropriate reference material for the external security manager installed.

## System action:

The program continues.

## User response:

Review the definition of the external security manager profile *esmprof*.

---

**GOM0161E        HNDIUCV SET FAILED *nn***

## Explanation

An attempt to establish an IUCV interrupt processing routine failed. The HNDIUCV SET system call returned code *nn*.

## System action

The Operations Manager command terminates.

## User response

See HNDIUCV in the *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

**GOM0162E        HNDIUCV CLR FAILED *nn***

## Explanation

An attempt to terminate an IUCV interrupt processing routine failed. The HNDIUCV CLR system call returned code *nn*.

## System action

The Operations Manager command terminates.

## User response

See HNDIUCV in the *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

**GOM0163E IUCV CONNECT *userid* FAILED *nn***

---

## Explanation

An attempt to establish an IUCV connection to *userid* failed. The CMSIUCV CONNECT system call returned code *nn*.

## System action

The Operations Manager command terminates.

## User response

See CMSIUCV in the *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

**GOM0164E IUCV SEVER *userid* FAILED *nn***

---

## Explanation

An attempt to terminate an IUCV connection to *userid* failed. The CMSIUCV SEVER system call returned code *nn*.

## System action

The program continues.

## User response

See CMSIUCV in the *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

**GOM0165E IUCV SEVER *userid alias* FAILED, NAME NOT FOUND**

---

## Explanation

An internal lookup for *userid* failed. This is an internal logic error.

## System action

The program continues.

## User response

Contact IBM Technical Support for assistance.

---

**GOM0166E EXTERNAL INTERRUPT *data* IS NOT VALID**

---

## Explanation

An external interrupt with an incorrect *data* has been presented for processing.

## System action

The program continues.

## User response

Contact IBM Technical Support for assistance.

---

**GOM0167E IUCV ACCEPT *userid* FAILED *nn***

---

## Explanation

An attempt to establish an IUCV connection with *userid* failed. The CMSIUCV ACCEPT system call returned code *nn*.

## System action

The program continues.

## User response

See CMSIUCV in the *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

**GOM0168E CONNECTION COMPLETE FOR PATHID *pppp* FAILED, PATH NOT FOUND**

---

## Explanation

An internal lookup for IUCV PATHID, represented by *pppp*, failed. This is an internal logic error.

## System action

The program continues.

## User response

Contact IBM Technical Support for assistance.

---

**GOM0169E IUCV SEVER PATHID *pppp* FAILED *nn***

---

## Explanation

An attempt to terminate an IUCV connection with PATHID *pppp* failed. The CMSIUCV SEVER system call returned code *nn*.

## System action

The program continues.

## User response

See CMSIUCV in the *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

<b>GOM0170E</b>	<b>IUCV SEVER PATHID <i>pppp</i> FAILED, PATH NOT FOUND</b>
-----------------	---

---

## Explanation

An internal lookup for the IUCV PATHID, which is represented by *pppp*, failed. This is an internal logic error.

## System action

The program continues.

## User response

Contact IBM Technical Support for assistance.

---

<b>GOM0171E</b>	<b>IUCV RECEIVE FOR <i>userid alias</i> <i>pppp</i> FAILED <i>nn</i></b>
-----------------	--

---

## Explanation

An attempt to receive an IUCV message from *userid* failed. The IUCV RECEIVE system call returned with code *nn*. The connection was attempted for PATHID *pppp*.

## System action

The program continues.

## User response

See IUCV RECEIVE in *z/VM CP Programming Services* for a list of codes and descriptions.

---

<b>GOM0172E</b>	<b>IUCV RECEIVE FOR PATHID <i>pppp</i> FAILED, PATH NOT FOUND</b>
-----------------	---

---

## Explanation

An internal lookup for IUCV PATHID, which is represented by *pppp*, failed. This is an internal logic error.

## System action

The program continues.

## User response

Contact IBM Technical Support for assistance.

---

<b>GOM0173E</b>	<b>IUCV REPLY <i>userid alias pppp</i> FAILED <i>nn</i></b>
-----------------	---

---

## Explanation

An attempt to send an IUCV reply to *userid* on PATHID *pppp* failed. The IUCV SEND system call returned code *nn*.

## System action

The program continues.

## User response

See IUCV REPLY in *z/VM CP Programming Services* for a list of codes and descriptions.

---

<b>GOM0174E</b>	<b>IUCV REPLY FOR <i>userid alias</i> FAILED, PATH NOT FOUND</b>
-----------------	--

---

## Explanation

An internal lookup for *userid* failed. This resulted in an internal logic error.

## System action

The program continues.

## User response

Contact IBM Technical Support for assistance.

---

<b>GOM0175E</b>	<b>IUCV SEND <i>userid alias pppp</i> FAILED <i>nn</i></b>
-----------------	--

---

## Explanation

An attempt to send an IUCV message to *userid* on PATHID *pppp* failed. The IUCV SEND system call returned code *nn*.

## System action

The program continues.

## User response

See IUCV SEND in *z/VM CP Programming Services* for a list of codes and descriptions.

---

<b>GOM0176E</b>	<b>IUCV SEND FOR <i>userid alias</i> FAILED, PATH NOT FOUND</b>
-----------------	---

---

## Explanation

An internal lookup for *userid* failed. This resulted in an internal logic error.

## System action

The program continues.

## User response

Contact IBM Technical Support for assistance.

---

<b>GOM0177E</b>	<b>IUCV SEND COMPLETE FOR PATHID <i>pppp</i> FAILED, PATH NOT FOUND</b>
-----------------	---

---

## Explanation

An internal lookup for the IUCV PATHID, represented by *pppp*, failed. This is an internal logic error.

## System action

The program continues.

## User response

Contact IBM Technical Support for assistance.

---

<b>GOM0178E</b>	<b>IUCV SEND COMPLETE <i>userid</i> <i>alias pppp</i> FAILED, MSGID <i>mmmmmmmm</i> MSG NOT FOUND</b>
-----------------	---

---

## Explanation

An internal lookup for a message, represented by *mmmmmmmm*, failed. This is an internal logic error.

## System action

The program continues.

## User response

Contact IBM Technical Support for assistance.

---

<b>GOM0179E</b>	<b>IUCV SEND COMPLETE <i>userid alias</i> <i>pppp</i> FAILED <i>xxxxxxxx</i></b>
-----------------	--

---

## Explanation

The completion of an IUCV SEND to *userid* on PATHID *pppp* failed. The IPARML IPAUDIT field is represented by *xxxxxxxx*.

## System action

The program continues.

## User response

See IPARML IPAUDIT in *z/VM CP Programming Services* for a list of codes and descriptions.

---

<b>GOM0180E</b>	<b>FREE STORAGE SHORTAGE <i>userid</i> <i>alias pppp size code</i>, IUCV MSGS LOST</b>
-----------------	--

---

## Explanation

An attempt to process an IUCV message may cause a severe storage shortage. The IUCV message is ignored to avoid causing a severe storage shortage. One or more messages are ignored and are unable to be processed. If the source *userid* is a CP system service, the IUCV connection remains active. In other cases, the IUCV connection is terminated. The IUCV PATHID is represented by *pppp*. The amount of storage required to process the IUCV message is represented by *size*. The *code* is either the maximum contiguous available storage, or a return code from the CMSSTOR OBTAIN system call. See *z/VM CP Programming Services* for a list of CP system services.

## System action

The program continues.

## User response

Increase the amount of virtual storage allocated to the virtual machine running the Operations Manager component experiencing the storage shortage. The minimum amount of increase should be 50% of the existing VM storage size. If the problem persists after increasing the original size by 400%, contact IBM Technical Support for assistance.

---

<b>GOM0181E</b>	<b>IUCV CONNECT ATTEMPT BY <i>userid</i> NOT ALLOWED</b>
-----------------	--

---

## Explanation

An attempt was made to establish an IUCV connection to an Operations Manager component which isn't allowed to accept IUCV connections. The IUCV connection request from *userid* is rejected and the IUCV connection is terminated.

## System action

The program continues.

## User response

No action is required.

---

**GOM0183E IUCV CONNECTION TO SYSTEM  
SERVICE *service-name* SEVERED**

---

**Explanation:**

An IUCV connection to a CP system service was severed. See *z/VM CP Programming Services* for a list of CP system services..

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM0184L IUCV CONNECTING SYS=*sysid*  
VID=*vmid* ALI=*alias* USR=*userid*  
pppp**

---

**Explanation:**

Records an IUCV connection has been established. The *system*, *vmid*, *alias*, *userid*, and *pppp* (path ID) information are provided.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM0185L IUCV ACCEPTING SYS=*sysid*  
VID=*vmid* ALI=*alias* USR=*userid*  
pppp**

---

**Explanation:**

Records an IUCV connection has been established. The *system*, *vmid*, *alias*, *userid*, and *pppp* (path ID) information are provided.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM0186L IUCV DISCONNECTING SYS=*sysid*  
VID=*vmid* ALI=*alias* USR=*userid*  
pppp**

---

**Explanation:**

Records an IUCV connection has been terminated. The *system*, *vmid*, *alias*, *userid*, and *pppp* (path ID) information are provided.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM0187E IUCV SET CONNECTION USERID  
*userid alias* FAILED, PATH NOT  
FOUND**

---

**Explanation:**

An internal lookup for the IUCV connection for *userid* and *alias* failed. This is an internal logic error.

**System action:**

The program continues

**User response:**

Contact IBM Technical Support for assistance.

---

**GOM0188L IUCV SYSTEM AND USER SET  
SYS=*sysid* VID=*vmid* ALI=*alias*  
USR=*userid* pppp**

---

**Explanation:**

The IUCV connection represented by *vmid* and *alias* has established the system and user information represented by *sysid* and *userid*.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM0189E IUCV QUIECE RECVEIVED ON  
PATHID *pppp*, PATH NOT FOUND**

---

**Explanation:**

An internal lookup for the IUCV PATHID, represented by *pppp*, failed. This is an internal logic error.

**System action:**

The program continues.

**User response:**

Contact IBM Technical Support for assistance.

---

**GOM0190E IUCV QUIECE RECVEIVED ON  
PATHID *pppp*, VID=*vmid* ALI=*alias***

---

**Explanation:**

An IUCV QUIECE request was received on IUCV PATHID *pppp*.

**System action:**

The program continues.

**User response:**

Contact IBM Technical Support for assistance.

---

**GOM0191E IUCV RESUME RECEIVED ON  
PATHID *pppp*, PATH NOT FOUND**

---

**Explanation:**

An internal lookup for the IUCV PATHID, represented by *pppp*, failed. This is an internal logic error.

**System action:**

The program continues.

**User response:**

Contact IBM Technical Support for assistance.

---

**GOM0192E IUCV RESUME RECEIVED ON  
PATHID *pppp*, VID=*vmid* ALI=*alias***

**Explanation:**

An IUCV RESUME request was received on IUCV PATHID *pppp*.

**System action:**

The program continues.

**User response:**

Contact IBM Technical Support for assistance.

---

**GOM0193E IUCV SEND COMPLETE ERROR,  
PATHID *pppp* NOT FOUND**

**Explanation:**

An internal lookup for the IUCV PATHID, represented by *pppp*, has failed. This is an internal logic error.

**System action:**

The program continues.

**User response:**

Contact IBM Technical Support for assistance.

---

**GOM0194E IUCV SEND COMPLETE ERROR,  
PATHID *pppp* MSGID *nnnnnnnn*  
NOT FOUND**

**Explanation:**

An internal lookup for the IUCV PATHID/MSGID, represented by *pppp* and *nnnnnnnn*, has failed. This is an internal logic error.

**System action:**

The program continues.

**User response:**

Contact IBM Technical Support for assistance.

---

**GOM0195E IUCV MESSAGE ERROR, PATHID  
*pppp* NOT FOUND**

**Explanation:**

An internal lookup for the IUCV PATHID, represented by *pppp*, has failed. This is an internal logic error.

**System action:**

The program continues.

**User response:**

Contact IBM Technical Support for assistance.

---

**GOM0201L "*cmd*" SYS=*sysid* VID=*vmid*  
ALI=*alias* USR=*userid* SRC=*source-*  
id CLS=*c***

**Explanation**

The command *cmd* was issued by *userid* at system *sysid*. The source virtual machine is represented by *vmid*. The internal source path is identified by *source-*

*id*. The message class of the command is represented by *c*.

**System action**

The program continues.

**User response**

No action is required.

---

**GOM0202E UNKNOWN COMMAND *cmd***

**Explanation**

An unknown command, represented by *cmd*, was encountered by Operations Manager.

**System action**

The program continues.

**User response:**

Correct the command and retry.

---

**GOM0203E *userid* IS NOT AUTHORIZED FOR  
*cmd***

**Explanation**

The *userid* has not been authorized to execute the *cmd* command.

**System action**

The program continues.

**User response**

No action is required.

---

**GOM0204I *text***

**Explanation**

A user has issued the LOGTEXT command to log an entry in the Operations Manager log.

**System action**

The program continues.

**User response**

No action is required.

---

**GOM0205E COMMAND "*cmd*" SYNTAX NOT  
VALID**



## Explanation

An error exists in the syntax of the input command *cmd*.

## System action

The program continues.

## User response:

Correct the command and retry.

---

**GOM0206L**      *output*

## Explanation

This message is used to log the output from the CP Operations Manager command.

## System action

The program continues.

## User response

No action is required.

---

**GOM0208E**      **SCAN FAILED *nn***

## Explanation

An attempt to build a command parameter list failed. The SCAN system call returned code *nn*.

## System action

The program continues.

## User response

See SCAN in the *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

**GOM0209E**      **COMMAND RC= *nn***

## Explanation

CMS or CP processing of a command returned a non-zero return code.

## System action

The program continues.

## User response

See either the *z/VM CP Commands and Utilities Reference* or the *z/VM CMS Commands and Utilities Reference* for a list of command return codes.

---

**GOM0210L**      *output*

## Explanation

This message is used to log the output from the CMS Operations Manager command.

## System action

The program continues.

## User response

No action is required.

---

**GOM0211E**      ***command name* ALREADY DEFINED**

## Explanation

An attempt was made to define a resource and the resource already exists. Command *command* attempted to define *name*.

## System action

The program continues.

## User response:

Correct the command and retry.

---

**GOM0212E**      ***command name* NOT FOUND**

## Explanation

An attempt was made to delete a resource and the resource does not exist. Command *command* attempted to delete *name*.

## System action

The program continues.

## User response:

Correct the command and retry.

---

**GOM0213E**      **CONFIGURATION FILE "*filename filetype* *filemode*" NOT FOUND**

## Explanation

The CONFIG command is unable to find the file specified.

## System action

The program continues.

## User response:

Correct the command and retry.

---

**GOM0214E**      **SERVICE MACHINE *svmid* IS PROCESSING**

## Explanation

An attempt was made to delete a busy service machine. The name of the service machine is *svmid*.

## System action

The program continues.

### User response:

If required, use FORCE Y to force deletion.

---

<b>GOM0216L</b>	<b><i>userid "msg" VID=vmid SRC=source-id CLS=c</i></b>
-----------------	---

---

## Explanation

The message *msg* was originated from *userid* and received by Operations Manager. The message was forwarded by the CP system service or VM identified by *vmid*. The internal source path is identified by *source-id*. CLS is the message class.

## System action

The program continues.

## User response

No action is required.

---

<b>GOM0217E</b>	<b>COMMAND <i>cmd</i> NOT VALID FROM <i>src</i></b>
-----------------	---

---

## Explanation

An attempt was made to execute an Operations Manager command from an invalid source, represented by *src*.

## System action

The program continues.

### User response:

Correct the command and retry.

---

<b>GOM0218L</b>	<b>USER <i>accessing-user</i> ISSUED COMMAND <i>cmd</i> TO <i>monitored-user</i></b>
-----------------	--

---

## Explanation

The user *accessing-user* is viewing the console of *monitored-user* and has issued the command *cmd*.

## System action

The program continues.

## User response

No action is required.

---

<b>GOM0219E</b>	<b>COMBINED VIEW COMMAND FORMAT ERROR (FROM <i>accessing- user</i>)</b>
-----------------	---

---

## Explanation

The user *accessing-user* is viewing a combined view and entered a command incorrectly. The command syntax should be *target userid* followed by the command, such as *USERA command*, to indicate which user of the combined view is the target of the command.

## System action

The program continues.

## User response

Correct the syntax and retry the command.

---

<b>GOM0220E</b>	<b>COMBINED VIEW COMMAND TARGET <i>target-user</i> NOT ELIGIBLE (FROM <i>accessing-user</i>)</b>
-----------------	--

---

## Explanation

The user *accessing-user* is viewing a combined view and entered a command from a combined view for *target-user* which is not in the combined view.

## System action

The program continues.

## User response

Specify a *target-user* that is in the combined view.

---

<b>GOM0221E</b>	<b>COMBINED VIEW COMMAND TARGET <i>target-user</i> NOT MONITORED (FROM <i>accessing- user</i>)</b>
-----------------	--

---

## Explanation

The user *accessing-user* is viewing a combined view and entered a command from a combined view for *target-user* which is not monitored by Operations Manager.

## System action

The program continues.

## User response

Setup *target-user* as an Operations Manager monitored user using the SET SECUSER or SET OBSERVER command.

---

**GOM0223I**      **USER *userid* ISSUED COMMAND *cmd***

## Explanation

The user *userid* is viewing the spool content list and issued the command *cmd*.

## System action

The program continues.

## User response

No action is required.

---

**GOM0225E**      **Operand *operand* SYNTAX NOT VALID**

## Explanation:

The operand listed could not be parsed.

## System action:

The program continues.

## User response:

Correct the command and retry.

---

**GOM0226E**      **OPERAND *operand* IS NOT VALID FOR COMMAND *command***

## Explanation:

The operand listed is not valid for the command entered.

## System action:

The program continues.

## User response:

Correct the command and retry.

---

**GOM0227E**      **OPERAND *operand* SPECIFIED MULTIPLE TIMES**

## Explanation:

The operand listed was entered multiple times.

## System action:

The program continues.

## User response:

Correct the command and retry.

---

**GOM0228I**      **RULE *ruleid* PROCESSING LIMIT EXCEEDED**

## Explanation:

The rule *ruleid* was defined with a LIMIT operand. The start of the rule's ACTION operand is being suppressed

because the rule's LIMIT operand count per interval is exceeded. The rule's ACTION operand will start when the LIMIT operand interval expires.

## System action:

The program continues.

## User response

No action is required.

---

**GOM0229I**      **EVENT *emonid* PROCESSING LIMIT EXCEEDED**

## Explanation:

The event *emonid* was defined with a LIMIT operand. The start of the event's ACTION operand is being suppressed because the event's LIMIT operand count per interval is exceeded. The event's ACTION operand will start when the LIMIT operand interval expires.

## System action:

The program continues.

## User response

No action is required.

---

**GOM0230E**      **OPERAND VALUE FOR OPERAND *operand* SYNTAX NOT VALID**

## Explanation:

The operand value entered could not be processed. Possible reasons are the operand value is too long or short, the value is out of range, or improperly formatted.

## System action:

The program continues.

## User response:

Correct the command and retry.

---

**GOM0231E**      **REQUIRED OPERAND *operand* OMITTED**

## Explanation:

The operand indicated is required to process the command.

## System action:

The program continues.

## User response:

Correct the command and retry.

---

**GOM0232E**      **OPERAND *operand* MUTUALLY EXCLUSIVE WITH ANOTHER OPERAND**

## Explanation:

An operand entered is mutually exclusive with another operand entered for the command.

## System action:

The program continues.

**User response:**

Correct the command and retry.

---

**GOM0233E      AN OPTIONAL OPERAND IS REQUIRED**

**Explanation:**

One of the optional operands for the command is required.

**System action:**

The program continues.

**User response:**

Correct the command and retry.

---

**GOM0234E      LOG FILE FOR DATE *yyyy/mm/dd* NOT FOUND**

**Explanation:**

The log file for the requested date was not found.

**System action:**

The program continues.

**User response:**

Correct the command and retry.

---

**GOM0235L      *userid* data VID=*vmid* SRC=*source-id* CLS=*cc* TYP=*tt***

**Explanation:**

An event for *userid* of the class *cc* and type *tt* has occurred.

**System action:**

The event notification was forwarded by the CP system service or VM identified by *vmid*. The internal source path is identified by *source-id*. The data shown was provided in the event notification.

**User response**

No action is required.

---

**GOM0237E      APPL *applnm* DOES NOT SUPPORT COMMANDS**

**Explanation**

The IP application named *applnm* does not support interaction using the CMDIPCS command.

**System action**

The program continues.

**User response:**

Correct the command and retry.

---

**GOM0238E      REXX VARIABLE *varname* NOT LOCATED RC=*rc***

**Explanation**

The CMDIPCS command specified a REXXVAR *varname* operand which could not be located in the REXX environment.

**System action**

The program continues.

**User response:**

Determine why the REXX variable could not be located.

---

**GOM0240E      UNSUPPORTED COMMAND: *sysid vmid alias userid cmd-id cmd-lvl***

**Explanation:**

A message from another GOM component could not be processed. The IUCV connection information and command information are included.

**System action:**

The IUCV connection to the requesting Operations Manager component is terminated.

**User response:**

Determine why the Operations Manager components are unable to exchange messages. The most common cause is the two Operations Manager components are not at a compatible version or service level.

---

**GOM0241E      SUBCOM RESPONSE NOT RECEIVED**

**Explanation:**

A response to a SUBCOM command was not received within the allowed time frame.

**System action:**

The Operations Manager command terminates.

**User response:**

Contact IBM Technical Support for assistance.

---

**GOM0251E      CONFIGURATION FILE "*filename filetype filemode*" OPEN FAILED *rc***

**Explanation**

The CONFIG command is unable to open the specified file. The FSOPEN system call returned code *rc*.

**System action**

The program continues.

**User response**

See FSOPEN in the *z/VM CMS Macros and Functions Reference* for a list of return codes and descriptions.

---

**GOM0252E**      **CONFIGURATION FILE "filename  
filetype filemode" READ FAILED rc**

---

### Explanation

The CONFIG command is unable to read the specified file. The FSREAD system call returned code *rc*.

### System action

The program continues.

### User response

See FSREAD in the *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

**GOM0253E**      **CONFIGURATION FILE "filename  
filetype filemode" CLOSE FAILED  
nn**

---

### Explanation

The CONFIG command is unable to close the specified file. The FSCLOSE system call returned code *nn*.

### System action

The program continues.

### User response

See FSCLOSE in the *z/VM CMS Macros and Functions Reference* for a list of return codes and descriptions.

---

**GOM0254L**      **BEGIN FILE *fn ft fm* LAST UPDATE  
date**

---

### Explanation:

This message is issued before the contents of a file that contains Operations Manager commands is processed. The file name, file type, file mode and the last modified date of the files are listed.

### System action:

The program continues.

### User response

No action is required.

---

**GOM0255L**      **END FILE "filename file type  
filemode"**

---

### Explanation:

This message is issued after the contents of a file that contains Operations Manager commands is processed. The file name, file type, and file mode are listed.

### System action:

The program continues.

### User response

No action is required.

---

**GOM0260I**      **trigger-id name ACTION actnid  
TRIGGERED BY userid**

---

### Explanation

An action was triggered by a RULE, SCHEDULE, or MONITOR. The *trigger-id* is the type of resource triggering the action. The *name* is the name of the RULE, SCHEDULE, or MONITOR triggering the action. The *actnid* is the name of the action triggered. The *userid* is the name of the console recording the triggering.

### System action

The program continues.

### User response

No action is required.

---

**GOM0261E**      **ACTION actnid NOT FOUND FOR  
RULE | SCHEDULE | MONITOR**

---

### Explanation

An attempt to locate the action *actnid* that was specified for the rule *ruleid*, schedule *schdid*, machine monitor *mmonid*, spool monitor *smonid*, or event monitor *emonid*, page monitor *pmonid* or idle monitor *imonid* was not successful.

### System action

The program continues.

### User response:

Determine why the action is not defined.

---

**GOM0262I**      **ACTION action-name BEGIN FOR  
userid SERVER service-machine**

---

### Explanation

The action, listed as *action-name*, is beginning execution on *server-machine*

### System action

The program continues.

### User response

No action is required.

---

**GOM0265E**      **UNABLE TO SEND WORK UNIT TO  
SERVER *service-machine alias***

### Explanation

An attempt was made to send a request to execute a command on a service machine. However, this attempt failed.

### System action

The program continues.

### User response:

Determine why the work unit cannot be sent.

---

**GOM0266E**      **SERVICE MACHINE *userid* NOT  
LOCATED**

### Explanation

The user specified by *userid* attempted to execute a GOMSVL command without the corresponding DEFSRVM configuration command.

### System action

The program continues.

### User response

No action is required.

---

**GOM0267I**      **ACTION *action-name* END RC=*nn*  
SERVER *service-machine* FREE  
STG=*below/above***

### Explanation

The *action-name* ended with return code *nn* on *service-machine*. The largest amounts of contiguous free storage available *below* 16MB and *above* 16MB are listed. The storage sizes reported pertain to the *service-machine*.

### System action

The program continues.

### User response

No action is required.

---

**GOM0268E**      **COMMAND BUILD EXCEEDED  
AVAILABLE SPACE**

### Explanation

While dynamically building a command for execution, all of the available build space was exhausted.

### System action

The program continues.

### User response:

Contact IBM Technical Support for assistance.

---

**GOM0269L**      **COMMAND *cmd***

### Explanation

This message is used to log the dynamically constructed command.

### System action

The program continues.

### User response

No action is required.

---

**GOM0270L**      ***output-text***

### Explanation

This message is used to log the command output.

### System action

The program continues.

### User response

No action is required.

---

**GOM0271E**      **UNABLE TO PROCESS ACTION  
*actnid*, NOT AUTHORIZED TO SET  
ALTUSER**

### Explanation:

The Operations Manager component, GOMMAIN, does not have the appropriate privilege class to issue `diagnose x'00D4'`. The action *actnid* listed in this message was defined with the ALTUSER operand, but GOMMAIN is unable to establish the alternate user.

### System action:

The action is flushed (not processed) with a simulated completion code of "-14".

### User response:

Determine if the ALTUSER operand is required for the action. If the ALTUSER is not needed, modify the corresponding DEFACTN statement. Determine if the privileged class assigned to GOMMAIN is appropriate. If it is not appropriate, modify the privileges and restart Operations Manager component, GOMMAIN.

---

**GOM0272W**      **ACTION PROCESSING SERVER  
*service-machine* DISCONNECTED  
FROM *local-machine***

**Explanation:**

The Operations Manager component GOMSVM has terminated the IUCV connection to component GOMMAIN. The VM running GOMSVM is listed as *service-machine*. The VM running GOMMAIN is listed as *local-machine*. The *service-machine* is no longer available to process actions.

**System action:**

The program continues.

**User response:**

Determine why *service-machine* disconnected the IUCV connection.

---

**GOM0273E      UNABLE TO SEND COMPLETION  
RESPONSE TO *vmid alias***


---

**Explanation:**

After completion of an action on an action processing server, the response was not able to be returned to the dispatcher. The IUCV information about the connection is listed.

**System action:**

The program continues.

**User response:**

Determine why the results could not be returned.

---

**GOM0274W      ACTION QUEUE COUNT OF *count*  
EXCEEDS THRESHOLD OF *limit***


---

**Explanation:**

The number of queued action requests has exceeded the threshold established by the DEFOPTN ACTNQCK operand.

**System action:**

The program continues.

**User response:**

Determine why the threshold is exceeded. Use the STATIS DETAIL SERVICE command to obtain the status of action processing machines. Use the STATUS DETAIL ACTIONQ command to obtain information about queued action requests. Use the STATUS DETAIL OPTION to obtain information about current options.

---

**GOM0275W      ACTION QUEUE AGE OF *age*  
EXCEEDS THRESHOLD OF *limit***


---

**Explanation:**

The oldest queued action request has exceeded the threshold established by the DEFOPTN ACTNQCK operand.

**System action:**

The program continues.

**User response:**

Determine why the threshold is exceeded. Use the STATIS DETAIL SERVICE command to obtain the status of action processing machines. Use the STATUS

DETAIL ACTIONQ command to obtain information about queued action requests. Use the STATUS DETAIL OPTION to obtain information about current options.

---

**GOM0276E      UNABLE TO PROCESS ACTION  
*actnid* SET ALTU/SECL *userid*/  
*seclabel* FAILED *rc***


---

**Explanation:**

The Operations Manager component GOMMAIN was unable to establish the requested alternate user ID *userid* and the SECLABEL *seclabel* using diagnose x'00D4'. The action *actnid* listed was defined with the ALTUSER operand, but GOMMAIN is unable to establish the alternate user. Diagnose x'00D4' returned return code *rc*.

**System action:**

The action is flushed (not processed) with a simulated completion code of -14.

**User response:**

See *z/VM CP Programming Services* DIAGNOSE x'00D4' for a list of codes and descriptions. Determine why GOMMAIN cannot establish the alternate user and correct the error.

---

**GOM0281I      SECUSER OR OBSERVER NOT SET  
FOR *sysid userid***


---

**Explanation**

A VIEWCON command requested data for a user that is not being monitored.

**System action**

The program continues.

**User response:**

If the user should be monitored, check the user's SECUSER or OBSERVER settings.

---

**GOM0282I      MAXIMUM OBSERVERS FOR *sysid*  
*userid* IN USE**


---

**Explanation**

A VIEWCON command requested data for a user that is being monitored by the maximum number of observers (other users).

**System action**

The program continues.

**User response**

No action is required.

---

**GOM0283I      UNABLE TO DELIVER DATA TO  
                  sysid userid**

---

**Explanation**

An attempt to send data to a user executing the VIEWLOG or VIEWCON command failed.

**System action**

The program continues.

**User response**

No action is required.

---

**GOM0285E      UNRECOVERABLE ERROR ON  
                  VIRTUAL PRINTER device-number**

---

**Explanation**

An error occurred while processing a VIEWCON or VIEWLOG command with the MODE RDR operand.

**System action**

The program continues.

**User response:**

Determine why the command encountered an error and correct the problem.

---

**GOM0286E      NO DATA TO DISPLAY**

---

**Explanation**

The current console or current log does not contain any lines within the DATERNG operand specified on the VIEWCON or VIEWLOG command.

**System action**

The program continues.

**User response**

No action is required.

---

**GOM0287E      ADDRESS SPACE CREATE FOR  
                  data-space FAILED rc**

---

**Explanation:**

An attempt to create a data space with the name *data-space* failed. The ADRSPACE CREATE system call returned code *rc*.

**System action:**

The program continues.

**User response:**

See ADRSPACE CREATE in *z/VM CP Programming Services* for a list of codes and descriptions.

---

**GOM0288E      ACCESS LIST ENTRY ADD FOR  
                  data-space alet FAILED rc**

---

**Explanation:**

An attempt to add an ALET to the AL to access *data-space* failed. The ALSERV ADD system call returned code *rc*.

**System action:**

The program continues.

**User response:**

See ALSERV ADD in *z/VM CP Programming Services* for a list of codes and descriptions.

---

**GOM0301E      CONSOLE OPEN FAILED rc**

---

**Explanation**

An attempt to open a console failed. The CONSOLE OPEN system call returned code *rc*.

**System action**

The Operations Manager command terminates.

**User response**

See CONSOLE in the *z/VM CMS Macros and Functions Reference* for a list of return codes and descriptions.

---

**GOM0302E      CONSOLE QUERY FAILED rc**

---

**Explanation**

An attempt to query a console failed. The CONSOLE QUERY system call returned code *rc*.

**System action**

The Operations Manager command terminates.

**User response**

See CONSOLE in the *z/VM CMS Macros and Functions Reference* for a list of return codes and descriptions.

---

**GOM0303E      CONSOLE DEVICE NOT  
                  SUPPORTED (hex-data)**

---

**Explanation**

The device being used when executing VIEWCON or VIEWLOG is not supported. The *hex-data* describes the console attributes.



## System action

The Operations Manager command terminates.

## User response:

Determine why the device is not supported. The device size minimum is 10x80. The device size maximum is 256 rows and 256 columns. The maximum device size (rows x cols) is 32K

---

**GOM0304E      CONSOLE WRITE FAILED *rc***

---

## Explanation

An attempt to write to a console failed. The CONSOLE WRITE system call returned code *rc*

## System action

The Operations Manager command terminates.

## User response

See CONSOLE in the *z/VM CMS Macros and Functions Reference* for a list of return codes and descriptions.

---

**GOM0305E      CONSOLE READ FAILED *rc***

---

## Explanation

An attempt to read a console failed. The CONSOLE READ system call returned code *rc*.

## System action

The Operations Manager command terminates.

## User response

See CONSOLE in the *z/VM CMS Macros and Functions Reference* for a list of return codes and descriptions.

---

**GOM0306E      RETURN CODE *rc* FROM *cmd***

---

## Explanation

The command *cmd* returned a non-zero return code *rc*.

## System action

The program continues.

## User response

No action is required.

---

**GOM0307E      CONSOLE INPUT FORMAT ERROR**

---

## Explanation:

The inbound console data stream contains unexpected or unsupported data.

## System action:

The program continues.

## User response:

Contact IBM Technical Support for assistance.

---

**GOM0308E      CONSOLE INPUT EXCEEDS  
MAXIMUM COMMAND LENGTH**

---

## Explanation:

The command input to VIEWLOG or VIEWCON exceeds the allowable length.

## System action:

The program continues.

## User response:

Contact IBM Technical Support for assistance.

---

**GOM0309E      FSOPEN FOR *filename filetype*  
*filemode* FAILED *rc***

---

## Explanation:

The VIEWLOG/VIEWCON command is unable to open the specified file. The FSOPEN system call returned code *rc*.

## System action:

The program continues.

## User response:

See FSOPEN in the *z/VM CMS Macros and Functions Reference* for a list of return codes and descriptions.

---

**GOM0310E      FSWRITE FOR *filename filetype*  
*filemode* FAILED *rc***

---

## Explanation:

The VIEWLOG/VIEWCON command is unable to write the specified file. The FSWRITE system call returned code *rc*.

## System action:

The program continues.

## User response:

See FSWRITE in the *z/VM CMS Macros and Functions Reference* for a list of return codes and descriptions.

---

**GOM0311E      FSCLOSE FOR *filename filetype*  
*filemode* FAILED *rc***

---

## Explanation:

The VIEWLOG/VIEWCON command is unable to close the specified file. The FSCLOSE system call returned code *rc*.

## System action:

The program continues.

## User response:

See FSCLOSE in the *z/VM CMS Macros and Functions Reference* for a list of return codes and descriptions.

---

**GOM0312E      FILE *filename filetype filemode*  
ALREADY EXISTS; SPECIFY  
REPLACE OPTION**

---

**Explanation:**

While retrieving the console and log information in a CMS file, an attempt was made to return information in an existing CMS file.

**System action:**

The program continues.

**User response:**

User needs to specify FILEOPT REPLACE option to replace the content of the specified file.

---

**GOM0351E      CONSOLE OPEN FAILED rc**

---

**Explanation**

An attempt to open a console failed. The CONSOLE OPEN system call returned code *rc*.

**System action**

Operations Manager terminates.

**User response**

See CONSOLE in the *z/VM CMS Macros and Functions Reference* for a list of return codes and descriptions.

---

**GOM0352E      CONSOLE QUERY FAILED rc**

---

**Explanation**

An attempt to query a console failed. The CONSOLE QUERY system call returned code *rc*.

**System action**

The Operations Manager command terminates.

**User response**

See CONSOLE in the *z/VM CMS Macros and Functions Reference* for a list of return codes and descriptions.

---

**GOM0353E      CONSOLE DEVICE NOT  
SUPPORTED hex-data**

---

**Explanation**

The device being used when executing VIEWSP is not supported. The *hex-data* describes the console attributes.

**System action**

The Operations Manager command terminates.

**User response:**

Determine why the device is not supported. The device size minimum is 10x80. The device size maximum is

256 rows and 256 columns. The maximum device size (rows x cols) is 32K

---

**GOM0354E      CONSOLE WRITE FAILED rc**

---

**Explanation**

An attempt to write to a console failed. The CONSOLE WRITE system call returned code *rc*.

**System action**

The Operations Manager command terminates.

**User response**

See CONSOLE in the *z/VM CMS Macros and Functions Reference* for a list of return codes and descriptions.

---

**GOM0355E      CONSOLE READ FAILED rc**

---

**Explanation**

An attempt to read a console failed. The CONSOLE READ system call returned code *rc*.

**System action**

The Operations Manager command terminates.

**User response**

See CONSOLE in the *z/VM CMS Macros and Functions Reference* for a list of return codes and descriptions.

---

**GOM0356E      SPOOL FILE NOT FOUND OR NOT  
VIEWABLE FOR USER userid FILE  
fileid**

---

**Explanation:**

The spool file requested was not found or is not a format viewable by VIEWSP.

**System action:**

The program continues.

**User response:**

Correct the command and retry.

---

**GOM0357I      UNRECOVERABLE ERROR ON  
VIRTUAL DEVICE device-number**

---

**Explanation:**

An error occurred while processing a VIEWSP command with the MODE RDR operand.

**System action:**

The program continues.

**User response:**

Determine why the command encountered an error and correct the problem.

---

**GOM0358W      SPOOL ACCESS FOR *data-1* CLASS  
                  *data-2* FAILED *data-3***

---

**Explanation:**

An attempt to access spool data failed. The data items list the error information.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM0359E      RETURN CODE *rc* FROM *cmd***

---

**Explanation**

The command *cmd* returned a non-zero return code *rc*.

**System action**

The program continues.

**User response**

No action is required.

---

**GOM0360E      CONSOLE INPUT FORMAT ERROR**

---

**Explanation:**

The inbound console data stream contains unexpected or unsupported data.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM0361E      CONSOLE INPUT EXCEEDS  
                  MAXIMUM COMMAND LENGTH**

---

**Explanation:**

The command input to VIEWLOG or VIEWCON exceeds the allowable length.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM0362E      UNABLE TO PROCESS SPOOL  
                  REQUEST (FILES: *file-max* TIME:  
                  *time-max*)**

---

**Explanation:**

An attempt to construct a list of spool files using the DIAGNOSE x'D8' system call has exceed either the

maximum number of files allowed or the maximum amount of time allowed.

**System action:**

The program continues.

**User response:**

Contact IBM Technical Support for assistance.

---

**GOM0363E      USER *userid* HAS SPOOL  
                  DUPLICATE SPID *nnnn* (SPFSYSID  
                  *sysfileid1 sysfileid2*)**

---

**Explanation:**

An attempt to construct a list of spool files using the DIAGNOSE x'D8' system call has detected the USERID, represented by *userid*, has multiple spool files with the same SPID, represented by *nnnn*. The system spool file IDs assigned to the files are represented by *sysfileid1* and *sysfileid2*.

**System action:**

The command terminates.

**User response:**

The duplicate spool file ID numbers for a single user should be eliminated. One method to eliminate the duplicates is to delete one of the entries. Another method is to transfer one of the files to a different user ID.

---

**GOM0401I      SPOOL USE: MONITOR *smonid*  
                  SPACE *nn* PERCENT, FILES *nn*  
                  PERCENT**

---

**Explanation**

A USAGE or INCREASE condition was encountered and the specified percentage of data space and file count was reached.

**System action**

The program continues.

**User response**

No action is required.

---

**GOM0402I      SPOOL CHG: MONITOR *smonid*  
                  SPACE *nn* PERCENT, FILES *nn*  
                  PERCENT**

---

**Explanation**

A USAGE or INCREASE condition was encountered and the specified percentage of data space and file count used has changed.

**System action**

The program continues.

## User response

No action is required.

---

<b>GOM0403I</b>	<b>SPOOL ALERT: MONITOR <i>smonid</i> type CONDITION REACHED</b>
-----------------	--

---

## Explanation

A USAGE or INCREASE condition was encountered for the specified spool usage monitor.

## System action

The program continues.

## User response

No action is required.

---

<b>GOM0404E</b>	<b>UNABLE TO COMPUTE SPOOL PERCENTAGES</b>
-----------------	--

---

## Explanation

An attempt was made to execute the required commands to monitor spool conditions and a failure occurred.

## System action

Operations Manager cannot monitor spool usage conditions.

## User response

See “Step 1. Update OPMGRM1” on page 5 for a list of privilege classes used by Operations Manager.

---

<b>GOM0405I</b>	<b>SPOOL MONITOR <i>smonid</i> PROCESSING LIMIT EXCEEDED</b>
-----------------	--

---

## Explanation:

The spool monitor *smonid* was defined with a LIMIT operand. The start of the spool monitor's ACTION operand is being suppressed because the LIMIT operand count per interval is exceeded. The spool monitor's ACTION operand will start when the LIMIT operand interval expires.

## System action:

The program continues.

## User response

No action is required.

---

<b>GOM0413E</b>	<b>FSOPEN FAILED <i>rc</i> FOR <i>filename</i> <i>filetype</i></b>
-----------------	--

---

## Explanation:

An attempt to open a file failed. The FSOPEN system call returned code *rc*.

## System action:

The program continues.

## User response:

See FSOPEN in *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

<b>GOM0414E</b>	<b>FSREAD FAILED <i>rc</i> FOR <i>filename</i> <i>filetype</i></b>
-----------------	--

---

## Explanation:

The attempt to read from a file failed. The FSREAD system call returned code *rc*.

## System action:

The program continues.

## User response:

See FSREAD in *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

<b>GOM0415E</b>	<b>FSWRITE FAILED <i>rc</i> FOR <i>filename</i> <i>filetype</i></b>
-----------------	---

---

## Explanation:

The attempt to write to a file failed. The FSWRITE system call returned code *rc*.

## System action:

The program continues.

## User response:

See FSWRITE in *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

<b>GOM0416E</b>	<b>FSCLOSE FAILED <i>rc</i> FOR <i>filename</i> <i>filetype</i></b>
-----------------	---

---

## Explanation

The attempt to close a file failed. The FSCLOSE system call returned code *rc*.

## System action

The program continues.

## User response:

See FSCLOSE in *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

<b>GOM0417I</b>	<b><i>req-type</i> REQUEST <i>reqid</i> FROM <i>userid</i> TO <i>ipcsid</i> BEGIN</b>
-----------------	---

---

## Explanation

The processing of the request *req-type* from *userid* to *ipcsid* is starting.

## System action

The program continues.

## User response

No action is required.

---

**GOM0418I**      *req-type* REQUEST *reqid* FROM  
*userid* TO *ipcsid* ENDED

## Explanation

The processing of the request *req-type* from *userid* to *ipcsid* is complete.

## System action

The program continues.

## User response

No action is required.

---

**GOM0419E**      *req-type* REQUEST *reqid* FROM  
*userid* TO *ipcsid* FAILED

## Explanation

The processing of the request *req-type* from *userid* to *ipcsid* encountered errors.

## System action

The program continues.

## User response:

Determine why the request could not complete without errors.

---

**GOM0420I**      *req-type* REQUEST *reqid* FROM  
*userid* TO *ipcsid* QUEUED

## Explanation

The processing of the request *req-type* from *userid* to *ipcsid* is queued for subsequent processing.

## System action

The program continues.

## User response

No action is required.

---

**GOM0421E**      UNABLE TO PARSE OR RESOLVE  
ADDRESS *addr*

## Explanation:

The format or value of *addr* is not a valid IPV4, IPV6, or resolvable DNS entry. If *addr* is a DNS entry, the resolution may require time to resolve.

## System action:

The program continues.

## User response:

Review *addr* for a valid IPV4, IPV6, or DNS entry.

---

**GOM0422E**      UNABLE TO PROCESS DNS REPLY:  
*ipcsid* hex-data

## Explanation:

A response from a DNS server contained an unexpected format or unexpected content.

## System action:

The program continues.

## User response:

Contact IBM Technical Support for assistance.

---

**GOM0423L**      *ipcsid* RESPONSE: *addr* TYPE *type*  
CLASS *class* CODE *code*

## Explanation:

The DNS resolution of *addr* from the IP connection *ipcsid* responded with the listed type, class, and code.

## System action:

The program continues.

## User response

No action is required.

---

**GOM0424L**      *ipcsid* ANSWER: *addr* TYPE *type*  
CLASS *class* TTL *time-to-live* LEN  
length DATA hex-data

## Explanation:

The DNS resolution of *addr* from the IP connection *ipcsid* responded with the listed type, class, time to live, and data.

## System action:

The program continues.

## User response

No action is required.

---

**GOM0425L**      *ipcsid* ALIAS: *addr* *alias*

## Explanation:

The DNS resolution of *addr* from the IP connection *ipcsid* responded with a CNAME response *alias*.

## System action:

The program continues.

## User response

No action is required.

---

**GOM0426L**      *ipcsid REFERRAL: addr DNS-server-name*

**Explanation:**

The DNS resolution of *addr* from the IP connection *ipcsid* responded with a referral to another DNS server.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM0427E**      *ipcsid UNSUPPORTED COMMAND: cmd-id cmd-lvl hex-data*

**Explanation:**

A message from another GOM component could not be processed.

**System action:**

The program continues.

**User response:**

Contact IBM Technical Support for assistance.

---

**GOM0428L**      *ipcsid vmid alias SEND: hex-data*

**Explanation:**

Tracing is active to obtain diagnostic data for DNS requests. The packet sent to the DNS server is listed.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM0429L**      *ipcsid vmid alias RECV: hex-data*

**Explanation:**

Tracing is active to obtain diagnostic data for DNS responses. The packet received from the DNS server is listed.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM0430E**      *ipcsid vmid alias UNSUPPORTED MESSAGE FORMAT FROM ipaddr*

**Explanation:**

The syslog data received cannot be processed. The message length field of the message is not numeric, or out of range.

**System action:**

The program continues.

**User response:**

Examine the DEFIPCS command for the session named *ipcsid* for the correct operands. Examine the configuration of the remote syslog transmitter. See “Collection and processing of syslog data” on page 200 for configuration considerations.

---

**GOM0431I**      **MACHINE MONITOR *mmonid* PROCESSING LIMIT EXCEEDED**

**Explanation:**

The machine monitor *mmonid* was defined with a LIMIT operand. The start of the machine monitor's ACTION operand is being suppressed because the machine monitor's LIMIT operand count per interval is exceeded. The machine monitor's ACTION operand will start when the LIMIT operand interval expires.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM0441I**      **SCHEDULE *schedid* PROCESSING LIMIT EXCEEDED**

**Explanation:**

The schedule *schedid* was defined with a LIMIT operand. The start of the schedule's ACTION operand is being suppressed because the schedule's LIMIT operand count per interval is exceeded. The schedule's ACTION operand will start when the LIMIT operand interval expires.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM0451I**      **PAGE USE: MONITOR *pmonid* SPACE *nn* PERCENT**

**Explanation:**

A USAGE or INCREASE condition was encountered and the specified percentage of page space was reached.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM0452I**      **PAGE CHG: MONITOR *pmonid* SPACE *nn* PERCENT**

**Explanation:**

A USAGE or INCREASE condition was encountered and the specified percentage of page space was reached.

**System action:**  
The program continues.

### User response

No action is required.

---

**GOM0453I PAGE ALERT: MONITOR *pmonid*  
type CONDITION REACHED**

---

**Explanation:**  
A USAGE or INCREASE condition was encountered for the specified page usage monitor.

**System action:**  
The program continues.

### User response

No action is required.

---

**GOM0454E UNABLE TO COMPUTE PAGE  
PERCENTAGES**

---

**Explanation:**  
An attempt was made to issue the required commands to monitor page conditions and a failure occurred.

**System action:**  
Operations Manager cannot monitor page usage conditions

**User response:**  
See “Step 1. Update OPMGRM1” on page 5 for a list of privilege classes used by Operations Manager.

---

**GOM0455I PAGE MONITOR *pmonid*  
PROCESSING LIMIT EXCEEDED**

---

**Explanation:**  
The page space monitor *pmonid* was defined with a LIMIT operand. The start of the page space monitor's ACTION operand is being suppressed because the page space monitor's LIMIT operand count per interval is exceeded. The page space monitor's ACTION operand will start when the LIMIT operand interval expires.

**System action:**  
The program continues.

### User response

No action is required.

---

**GOM0461I IDLE MONITOR *imonid*  
PROCESSING LIMIT EXCEEDED**

---

**Explanation:**  
The idle monitor *imonid* was defined with a LIMIT operand. The start of the idle monitor's ACTION operand is suppressed because the idle monitor's LIMIT operand count per interval is exceeded. The

idle monitor's ACTION operand will start with the LIMIT operand interval ends.

**System action:**  
The program continues.

### User response

No action is required.

---

**GOM0462E IDLE MONITOR *imonid* resource-  
type resource-id NOT FOUND**

---

**Explanation:**  
The idle monitor *imonid* could not locate the recourse to monitor.

**System action:**  
Operations Manager cannot monitor the resource activity.

**User response:**  
Define the resource to monitor.

---

**GOM0471W FREE STORAGE FOR *server-name*  
IS *nn*MB AND BELOW THE  
THRESHOLD OF *nn* MB**

---

**Explanation:**  
The amount of contiguous free storage for *server-name* has dropped below the threshold established by the DEFOP TN FSTORCK operand.

**System action:**  
The program continues.

**User response:**  
Increase the amount of virtual storage allocated to the virtual machine *server-name*. The minimum amount of increase should be 50% of the existing VM storage size. If the problem persists after increasing the original size by 400%, contact IBM Technical Support for assistance. Determine why the threshold is exceeded. Use the STATIS DETAIL SERVICE command to obtain the status of action processing machines. Use the STATUS DETAIL OPTION to obtain information about current options.

---

**GOM0501E IP OPERATION *type* TO *ipcsid*  
SOCK *xxxx* FAILED AT OFFSET  
*xxxxxxx***

---

**Explanation:**  
An IP operation failed to start. The operations type, the name of the IP connection, the socket number, and offset in the module are listed.

**System action:**  
The program continues.

**User response:**  
Determine the cause of the error and correct the problem.

---

**GOM0502E**      **IP OPERATION type TO *ipcsid*  
SOCK *xxxx* FAILED *rc* AT OFFSET  
*xxxxxxx***

---

**Explanation:**

An IP operation did not complete correctly. The operation type, the name of the IP connection, the socket number, the return code, and offset in the module are listed.

**System action:**

The program continues.

**User response:**

Determine the cause of the error and correct the problem.

---

**GOM0503E**      ***ipcsid* PARM FIELD CONTENTS  
NOT VALID**

---

**Explanation:**

The contents of the PARM field are not valid. One or more of the PARM field values are not correct.

**System action:**

The program continues.

**User response:**

Correct the error in the PARM field and redefine the session.

---

**GOM0899I**      **OPERATIONS MANAGER V1R6  
BUILT *yyyy/mm/dd hh:mm***

---

**Explanation**

The STATUS command outputs various information about the current processing environment. This message tells you the date and time of the Operations Manager build you are running.

**System action**

The program continues.

**User response**

No action is required.

---

**GOM0900I**      ***nn* RULE(S) DEFINED**

---

**Explanation**

The STATUS command outputs various information about the current processing environment. This message tells you how many rules are defined.

**System action**

The program continues.

**User response**

No action is required.

---

**GOM0902I**      ***nn* MACHINE MONITOR(S)  
DEFINED**

---

**Explanation**

The STATUS command outputs various information about the current processing environment. This message tells you how many monitored virtual machines are defined.

**System action**

The program continues.

**User response**

No action is required.

---

**GOM0903I**      ***nn* SCHEDULE(S) DEFINED**

---

**Explanation**

The STATUS command outputs various information about the current processing environment. This message tells you how many schedules are defined.

**System action**

The program continues.

**User response**

No action is required.

---

**GOM0904I**      ***nn* ACTION(S) DEFINED**

---

**Explanation**

The STATUS command outputs various information about the current processing environment. This message tells you how many actions are defined.

**System action**

The program continues.

**User response**

No action is required.

---

**GOM0905I**      ***nn* ACTION(S) SCHEDULED FOR  
PROCESSING**

---



## Explanation

The STATUS command outputs various information about the current processing environment. This message indicates how many actions are scheduled for processing.

## System action

The program continues.

## User response

No action is required.

---

**GOM0906I      nn GROUP(S) DEFINED**

---

## Explanation

The STATUS command outputs various information about the current processing environment. This message indicates how many groups are defined.

## System action

The program continues.

## User response

No action is required.

---

**GOM0907I      nn DATE(S) DEFINED**

---

## Explanation

The STATUS command outputs various information about the current processing environment. This message tells you how many holidays are defined.

## System action

The program continues.

## User response

No action is required.

---

**GOM0908I      nn SPOOL MONITOR(S) DEFINED**

---

## Explanation

The STATUS command outputs various information about the current processing environment. This message indicates how many spool monitors are defined.

## System action

The program continues.

## User response

No action is required.

---

**GOM0909I      nn EVENT MONITOR(S) DEFINED**

---

## Explanation:

The STATUS command outputs various information about the current processing environment. This message indicates how many event monitors are defined.

## System action:

The program continues.

## User response

No action is required.

---

**GOM0910I      nn PAGE MONITOR(S) DEFINED**

---

## Explanation:

The STATUS command outputs various information about the current execution environment. This message indicates how many page space monitors are defined.

## System action:

The program continues.

## User response

No action is required.

---

**GOM0911I      nn IDLE MONITOR(S) DEFINED**

---

## Explanation:

The STATUS command outputs various information about the current processing environment. This message indicates how many idle monitors are defined.

## System action:

The program continues.

## User response

No action is required.

---

**GOM0920I      nn SERVICE MACHINE(S) DEFINED**

---

## Explanation

The STATUS command outputs various information about the current processing environment. This message tells you how many service machines are defined.

## System action

The program continues.

## User response

No action is required.

---

**GOM0921I**      *nn* **USER(S)/GROUP(S)**  
**AUTHORIZED**

## Explanation

The STATUS command outputs various information about the current processing environment. This message tells you how many users and groups were granted authority.

## System action

The program continues.

## User response

No action is required.

---

**GOM0922I**      *nn* **IUCV CONNECTION(S)**

## Explanation

The STATUS command outputs various information about the current processing environment. This message indicates the number of IUCV connections.

## System action

The program continues.

## User response

No action is required.

---

**GOM0923I**      *nn* **DATA SPACE SIZE**  
**DEFINITION(S)**

## Explanation

The STATUS command outputs various information about the current processing environment. This message tells you how many data space definitions are defined.

## System action

The program continues.

## User response

No action is required.

---

**GOM0924I**      *nn* **COMBINED VIEW(S)**

## Explanation

The STATUS command outputs various information about the current processing environment. This message tells you how many combined views are defined.

## System action

The program continues.

## User response

No action is required.

---

**GOM0925I**      *nn* **USER CONSOLE(S) RECORDING**

## Explanation

The STATUS command outputs various information about the current processing environment. This message tells you how many user consoles are recording.

## System action

The program continues.

## User response

No action is required.

---

**GOM0926I**      *nn* **IP SESSION(S) DEFINED**

## Explanation:

The STATUS command outputs various information about the current processing environment. This message tells you how many IP communication sessions are defined.

## System action:

The program continues.

## User response

No action is required.

---

**GOM0940I**      **RULE *rule-name* TRIGGERED *nn***  
**TIME(S)**

## Explanation

The STATUS command outputs various information about the current processing environment. This message tells you how many times the rule *rule-name* was triggered.

## System action

The program continues.

## User response

No action is required.

---

**GOM0942I**      **MACHINE MONITOR *mmonid***  
**TRIGGERED *nn* TIME(S)**

## Explanation

The STATUS command outputs various information about the current processing environment. This message tells you how many times the monitored virtual machine was triggered.

## System action

The program continues.

## User response

No action is required.

---

**GOM0943I**      **SCHEDULE *schdid* TRIGGERED *nn***  
**TIME(S)**

## Explanation

The STATUS command outputs various information about the current processing environment. This message tells you how many times the schedule *schdid* was triggered.

## System action

The program continues.

## User response

No action is required.

---

**GOM0944I**      **ACTION *actnid* PROCESSED *nn***  
**TIME(S)**

## Explanation

The STATUS command outputs various information about the current processing environment. This message tells you how many times the action *actnid* was processed.

## System action

The program continues.

## User response

No action is required.

---

**GOM0945I**      **ACTION *actnid* SCHEDULED**  
**FOR PROCESSING (*console-***

***id/name/service-machine/service-***  
***system/token*)**

## Explanation

The STATUS command outputs various information about the current processing environment. This message tells you where the action is scheduled for processing and what triggered the action.

## System action

The program continues.

## User response

No action is required.

---

**GOM0946I**      **GROUP *groupid* INCLUDES USER |**  
**GROUP *userid* | *groupid***

## Explanation

The STATUS command outputs various information about the current processing environment. This message tells you the user ID or group ID that is included in this group.

## System action

The program continues.

## User response

No action is required.

---

**GOM0947I**      **DATE *dateid* FROM *yyyy/mm/dd-***  
***hh:mm* TO *yyyy/mm/dd-hh:mm***

## Explanation

The STATUS command outputs various information about the current processing environment. This message tells you the date and time range for the defined holiday.

## System action

The program continues.

## User response

No action is required.

---

**GOM0948I**      **SPOOL MONITOR *smonid***  
**TRIGGERED *nn* TIME(S)**

## Explanation

The STATUS command outputs various information about the current processing environment. This message tells you how many times the specified spool monitor was triggered.

## System action

The program continues.

## User response

No action is required.

---

**GOM0949I**      **EVENT MONITOR *emonid***  
**TRIGGERED *nn* TIME(S)**

---

### Explanation:

The STATUS command outputs various information about the current processing environment. This message tells you how many times an event monitor was triggered.

### System action:

The program continues.

## User response

No action is required.

---

**GOM0950I**      **PAGE MONITOR *pmonid***  
**TRIGGERED *nn* TIME(S)**

---

### Explanation:

The STATUS command outputs various information about the current processing environment. This message tells you how many times a page space monitor was triggered.

### System action:

The program continues.

## User response

No action is required.

---

**GOM0951I**      **IDLE MONITOR *imonid***  
**TRIGGERED *nn* TIME(S)**

---

### Explanation:

The STATUS command outputs various information about the current processing environment. This message indicates how many times an idle monitor was triggered.

### System action:

The program continues.

## User response

No action is required.

---

**GOM0960I**      **SERVICE MACHINE *srvmid***  
**PROCESSED *nn* COMMAND(S)**

---

## Explanation

The STATUS command outputs various information about the current processing environment. This message tells you how many commands the specified service machine has processed.

## System action

The program continues.

## User response

No action is required.

---

**GOM0962I**      **IUCV *sysid vmid alias userid***  
**PATHID *pathid* MSGS *nnn***

---

### Explanation:

The STATUS command outputs various information about the current processing environment. This message tells you about IUCV connection information for *sysid*, *userid*, *alias*, *pathid*, the allowed number of outstanding messages, and the count of messages exchanged.

### System action:

The program continues.

## User response

No action is required.

---

**GOM0965I**      **CONSOLE *userid-sysid* ASIT**  
**XXXXXXXXXXXXXXXXX ALET XXXXXXXX**  
**SIZE *nn* M MSGS *msg-count***

---

### Explanation:

The STATUS command outputs various information about the current processing environment. This message tells you about consoles being recorded.

### System action:

The program continues.

## User response

No action is required.

---

**GOM0966I**      **NAME *ipcsid* USER *vmid alias* APPL**  
***applid* SOCKET *nn***

---

### Explanation:

The STATUS command outputs various information about the current processing environment. This message tells you about IP connections and processing applications.

### System action:

The program continues.

### User response

No action is required.

---

**GOM0970I      USER *userid* SPOOL FILE ID *fileid*  
IS USING *nn* SPOOL BLOCKS**

---

### Explanation

The STATUS command outputs various information about the current processing environment. This message tells you the number of spool blocks used by a file belonging to the specified user.

### System action

The program continues.

### User response

No action is required.

---

**GOM0971I      USER *userid* HAS *nn* SPOOL FILES  
USING *nn* SPOOL BLOCKS**

---

### Explanation

The STATUS command outputs various information about the current processing environment. This message tells you how many spool files and spool blocks for the specified user.

### System action

The program continues.

### User response

No action is required.

---

**GOM1050E      ADDRESS SPACE QUERY FAILED *rc***

---

### Explanation

An address space query failed. The ADRSPACE QUERY system call returned code *rc*.

### System action

The Operations Manager command terminates.

### User response

See ADRSPACE QUERY in the *z/VM CP Programming Services* documentation for a list of return codes and descriptions.

---

**GOM1051E      NUCLEUS DROP FAILED *rc***

---

### Explanation

An attempt to drop a nucleus extension failed.

### System action

The Operations Manager command terminates.

### User response

See NUCXDR0P in the *z/VM CMS Commands and Utilities Reference* for a list of return codes and descriptions.

---

**GOM1052E      NUCLEUS LOAD FAILED *rc***

---

### Explanation

An attempt to load a nucleus extension failed.

### System action

The Operations Manager command terminates.

### User response

See NUCXLOAD in the *z/VM CMS Commands and Utilities Reference* for a list of return codes and descriptions.

---

**GOM1053E      NUCLEUS QUERY FAILED *rc***

---

### Explanation

An attempt to query a nucleus extension failed.

### System action

The Operations Manager command terminates.

### User response

See NUCEXT QUERY in the *z/VM CMS Macros and Functions Reference* for a list of return codes and descriptions.

---

**GOM1054E      EXTERNAL HANDLER SET FAILED *rc***

---

### Explanation

An attempt to establish an external interrupt processing routine failed. The HNDEXT SET system call returned code *rc*.

### System action

The Operations Manager command terminates.

## User response

See HNDEXT in the *z/VM CMS Macros and Functions Reference* for a list of return codes and descriptions.

---

**GOM1055E      EXTENDED PARAMETER LIST NOT AVAILABLE**

## Explanation

When an Operations Manager command is started, it attempts to obtain the extended parameter list. The Operations Manager command was unable to obtain the extended parameter list.

## System action

The Operations Manager command terminates.

### User response:

Determine why the Operations Manager command is unable to obtain an extended parameter list.

---

**GOM1099S      UNRECOVERABLE  
ERROR ENCOUNTERED  
PSW=XXXXXXXXXXXXXXXXX CCCCCCCC  
rrrrrrr EP=XXXXXXXXX  
MCB=XXXXXXXXX**

## Explanation

An unrecoverable error condition was encountered.

## System action

The Operations Manager command terminates.

### User response:

Contact IBM Technical Support for assistance. See *z/VM CMS and REXX/VM Messages and Codes* for a list of codes and descriptions. The system code and reason code are represented by *ccccccc* and *rrrrrrr*.

---

**GOM1101E      ABNEXIT SET FAILED, RC=rc**

## Explanation

An attempt to establish an error recovery routine failed. The ABNEXIT SET system call returned code *rc*.

## System action

The Operations Manager command terminates.

### User response:

See ABNEXIT in the *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

**GOM1102E      ABNEXIT CLR FAILED, RC=rc**

## Explanation

An attempt to terminate an error recovery routine failed. The ABNEXIT CLR system call returned code *rc*.

## System action

The Operations Manager command terminates.

## User response

See ABNEXIT in the *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

**GOM1111E      LOGICAL DEVICE TABLE FULL**

## Explanation

Too many sessions are active. No new sessions are allowed until some sessions are ended.

## System action

The program continues.

## User response

End a session.

---

**GOM1112E      LOGICAL DEVICE INIT FAILED  
CC=cc RC=rc**

## Explanation

A DIAG x'7C' failure has occurred.

## System action

The program continues.

## User response

See Diagnose x'7C' in the *z/VM CP Programming Services* documentation for a list of return codes and descriptions.

---

**GOM1113E      CMSSTACK FAILED rc**

## Explanation

A CMSSTACK operation failed. The CMSSTACK system call returned code *rc*.

## System action

The program continues.

## User response

See CMSSTACK in the *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

<b>GOM1114E</b>	<b>CONSOLE OPEN FAILED <i>rc</i>, TRACE IGNORED</b>
-----------------	---

---

## Explanation

An attempt to open a console failed. The CONSOLE OPEN system call returned code *rc*.

## System action

The program continues.

## User response

See CONSOLE in the *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

<b>GOM1121E</b>	<b>HANDLE NOT FOUND</b>
-----------------	-------------------------

---

## Explanation

An attempt to TERM a session failed because the specified handle was not found.

## System action

The program continues.

## User response:

Correct the command and retry.

---

<b>GOM1122E</b>	<b>LOGICAL DEVICE TERM FAILED CC=<i>cc</i> RC=<i>rc</i></b>
-----------------	---

---

## Explanation

A DIAG x'7C' failure has occurred.

## System action

The program continues.

## User response

See Diagnose x'7C' in the *z/VM CP Programming Services* documentation for a list of return codes and descriptions.

---

<b>GOM1123E</b>	<b>CONSOLE CLOSE FAILED <i>rc</i></b>
-----------------	---------------------------------------

---

## Explanation

An attempt to close a console failed. The CONSOLE CLOSE system call returned code *rc*.

## System action

The program continues.

## User response

See CONSOLE in the *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

<b>GOM1131E</b>	<b>HANDLE NOT FOUND</b>
-----------------	-------------------------

---

## Explanation

An attempt to QUERY a session failed because the specified handle was not found.

## System action

The program continues.

## User response:

Correct the command and retry.

---

<b>GOM1132E</b>	<b>CMSSTACK FAILED <i>rc</i></b>
-----------------	----------------------------------

---

## Explanation

A CMSSTACK operation failed. The CMSSTACK system call returned code *rc*.

## System action

The program continues.

## User response

See CMSSTACK in the *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

<b>GOM1133E</b>	<b>FSOPEN FAILED <i>rc</i></b>
-----------------	--------------------------------

---

## Explanation

An attempt to open a file failed. The FSOPEN system call returned code *rc*.

## System action

The program continues.

## User response

See FSOPEN in the *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

<b>GOM1134E</b>	<b>FSWRITE FAILED <i>rc</i></b>
-----------------	---------------------------------

---

## Explanation

An attempt to write to a file failed. The FSWRITE system call returned code *rc*.

### System action

The program continues.

### User response

See FSWRITE in the *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

**GOM1135E FSCLOSE FAILED rc**

### Explanation

An attempt to close a file failed. The FSCLOSE system call returned code *rc*.

### System action

The program continues.

### User response

See FSCLOSE in the *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

**GOM1141E HANDLE NOT FOUND**

### Explanation

An attempt to INPUT a session failed because the specified handle was not found.

### System action

The program continues.

### User response:

Correct the command and retry.

---

**GOM1142E CONSOLE WRITE FAILED rc**

### Explanation

An attempt to write to a console failed. The CONSOLE WRITE system call returned code *rc*.

### System action

The Operations Manager command terminates.

### User response

See CONSOLE in the *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

**GOM1151E LOGICAL DEVICE WAS TERMINATED**

### Explanation

The session for a logical device was terminated by CP.

### System action

The program continues.

### User response

No action is required.

---

**GOM1152E LOGICAL DEVICE ACCEPT FAILED  
CC=cc RC=rc**

### Explanation

A DIAG x'7C' failure has occurred.

### System action

The program continues.

### User response

See Diagnose x'7C' in the *z/VM CP Programming Services* documentation for a list of codes and descriptions.

---

**GOM1153E LOGICAL DEVICE STATUS FAILED  
CC=cc RC=rc**

### Explanation

A DIAG x'7C' failure has occurred.

### System action

The program continues.

### User response

See Diagnose x'7C' in the *z/VM CP Programming Services* documentation for a list of codes and descriptions.

---

**GOM1154E WRITE CCW OPERATION CODE xx  
NOT VALID**

### Explanation

A CCW with operation code *xx* was received. The code *xx* was not recognized code.

### System action

The program continues.

### User response:

Contact IBM Technical Support for assistance.

---

**GOM1155E CONSOLE WRITE FAILED rc**



## Explanation

An attempt to write to a console failed. The CONSOLE WRITE system call returned code *rc*.

## System action

The program continues.

## User response

See CONSOLE in the *z/VM CMS Macros and Functions Reference* for a list of codes and descriptions.

---

**GOM1156E      LOGICAL DEVICE PRESENT  
                 FAILED CC=*cc* RC=*rc***

---

## Explanation

A DIAG *x'7C'* failure has occurred.

## System action

The program continues.

## User response

See Diagnose *x'7C'* in the *z/VM CP Programming Services* documentation for a list of codes and descriptions.

---

**GOM1157E      STRUCTURED FIELD ERRORS  
                 SF=*data***

---

## Explanation

An unsupported structured field was encountered. The field in error is displayed.

## System action

The program continues.

### User response:

Contact IBM Technical Support for assistance.

---

**GOM1158E      DATA STREAM ERROR DATA=*data***

---

## Explanation

An unsupported data stream was encountered. The data stream in error is displayed.

## System action

The program continues.

### User response:

Contact IBM Technical Support for assistance.

---

**GOM1181E      LOGICAL DEVICE PRESENT  
                 FAILED CC=*cc* RC=*rc***

---

## Explanation

A DIAG *x'7C'* failure has occurred.

## System action

The program continues.

## User response

See Diagnose *x'7C'* in the *z/VM CP Programming Services* documentation for a list of codes and descriptions.

---

**GOM2886I      SDFCMD Y/N**

---

## Explanation

The STATUS command outputs various information about the current processing environment. This message lists the current setting for DEFOPTN SDFCMD.

### System action:

The program continues.

### User response:

No action is required.

---

**GOM2887I      VCSCGRP *groupid***

---

## Explanation

The STATUS command outputs various information about the current processing environment. This message lists the name of the group whose members support receiving console input from their simulated hardware console. If a monitored guest is a member of the group, VIEWCON uses the guest's simulated hardware console instead of the guest's virtual console to send commands.

### System action:

The program continues.

### User response:

No action is required.

---

**GOM2888I      ESMCLASS *esmclass***

---

## Explanation

The STATUS command outputs various information about the current processing environment. This message lists the name of the ESM class containing the profiles used by Operations Manager.

### System action:

The program continues.

**User response:**

No action is required.

---

**GOM2889I      AUTHEXT extensions**


---

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the usage of the various ESM authorization extensions.

**System action:**

The program continues.

**User response:**

No action is required.

---

**GOM2890I      COMBINED VIEW *viewid*  
INCLUDES GROUP *groupid***


---

**Explanation**

The STATUS command outputs various information about the current processing environment. This message tells you about combined views to contain console output for users contained in a group.

**System action:**

The program continues.

**User response:**

No action is required.

---

**GOM2891I      COMBINED VIEW *viewid*  
INCLUDES USER *userid***


---

**Explanation**

The STATUS command outputs various information about the current processing environment. This message tells you about combined views to contain console output for users.

**System action:**

The program continues.

**User response:**

No action is required.

---

**GOM2892I      DATA SPACE CONSOLE FOR  
GROUP *groupid* SIZE *nn* M**


---

**Explanation**

The STATUS command outputs various information about the current processing environment. This message tells you about defined data space size definitions to contain console output for users or combined views contained in a group.

**System action:**

The program continues.

**User response:**

No action is required.

---

**GOM2893I      DATA SPACE CONSOLE FOR USER  
*userid* SIZE *nn* M**


---

**Explanation**

The STATUS command outputs various information about the current processing environment. This message tells you about defined data space size definitions to contain console output for users or combined views.

**System action:**

The program continues.

**User response:**

No action is required.

---

**GOM2894I      PROCESSING LIMIT THRESHOLD  
ACTION *actnid***


---

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the name of the action scheduled for execution when the LIMIT value of a rule, schedule, or monitor is reached.

**System action:**

The program continues.

**User response:**

No action is required.

---

**GOM2895I      IMONLIM *count,interval[,lmtactn]***


---

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the current DEFOPTN IMONLIM values for count, interval, and limit action. See [“DEFOPTN” on page 57](#) for information on setting these values.

**System action:**

The program continues.

**User response:**

No action is required.

---

**GOM2896I      PMONLIM *count,interval[,lmtactn]***


---

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the current DEFOPTN PMONLIM values for count, interval, and limit action. See [“DEFOPTN” on page 57](#) for information on setting these values.

**System action:**

The program continues.

**User response:**

No action is required.

---

**GOM2897I      EMONLIM *count,interval[,lmtactn]***


---

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the current DEFOPTN EMONLIM values for count, interval, and limit action. See “DEFOPTN” on page 57 for information on setting these values.

**System action:**

The program continues.

**User response:**

No action is required.

---

**GOM2898I SMONLIM count, interval[, lmtactn]**
**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the current DEFOPTN SMONLIM values for count, interval, and limit action. See “DEFOPTN” on page 57 for information on setting these values.

**System action:**

The program continues.

**User response:**

No action is required.

---

**GOM2899I SCHDLIM count, interval[, lmtactn]**
**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the current DEFOPTN SCHDLIM values for count, interval, and limit action. See “DEFOPTN” on page 57 for information on setting these values.

**System action:**

The program continues.

**User response:**

No action is required.

---

**GOM2900I MMONLIM count, interval[, lmtactn]**
**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the current DEFOPTN MMONLIM values for count, interval, and limit action. See “DEFOPTN” on page 57 for information on setting these values.

**System action:**

The program continues.

**User response:**

No action is required.

---

**GOM2901I RULELIM count, interval[, lmtactn]**
**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the current DEFOPTN RULELIM values for count, interval, and limit action. See “DEFOPTN” on page 57 for information on setting these values.

**System action:**

The program continues.

**User response:**

No action is required.

---

**GOM2902I FSTORCK interval, lvm-minsize, svm-minsize**
**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the current value for the DEFOPTN FSTORCK operand.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2903I ACTNQCK interval, count, age**
**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the current value for the DEFOPTN ACTNQCK operand.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2904I ELIGIBLE FOR PROCESSING AT date**
**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the date and time when a queued action block is available for processing.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2905I EXCLUDE SYSTEM GROUP exsysgrp**
**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the system group excluded from processing.

**System action:**

The program continues.

## User response

No action is required.

---

### GOM2906I      SYSTEM GROUP *sysgrp*

#### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the system group allowed for processing.

#### System action:

The program continues.

## User response

No action is required.

---

### GOM2907I      IPV6 ADDRESS *addr* SCOPE *nn*

#### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the IP connection address and scope.

#### System action:

The program continues.

## User response

No action is required.

---

### GOM2908I      IPV4 ADDRESS *addr*

#### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the IP connection address.

#### System action:

The program continues.

## User response

No action is required.

---

### GOM2909I      PORT *nn*

#### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the IP connection port.

#### System action:

The program continues.

## User response

No action is required.

---

### GOM2910I      ADDR *addr*

#### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the address value from the IP session definition.

#### System action:

The program continues.

## User response

No action is required.

---

### GOM2911I      PARM *parm*

#### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the parm value from the IP session definition.

#### System action:

The program continues.

## User response

No action is required.

---

### GOM2921I      DELAY *nn* MINUTES *nn* SECONDS

#### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the amount of time to delay before the command of an action is processed.

#### System action:

The program continues.

## User response

No action is required.

---

### GOM2922I      PROCESSING ENV SYSTEM OVERRIDE *procsys*

#### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the processing environment system override used to process an action.

#### System action:

The program continues.

## User response

No action is required.

---

### GOM2923I      PROCESSING ENV SYSTEM CAN BE OVERRIDDEN

#### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the ability for an action's processing environment system to be changed.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2924I      PROCESSING ENV SYSTEM**  
*procenv*

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists an action's default processing environment system.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2925I      MESSAGE CLASS *cls***

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the message class required to trigger a rule.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2926I      STG TOTAL=*vmsize* FREE**  
**STG=*below/above***

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the storage information returned upon completion of an action, which is the total amount of storage defined and the maximum amount of contiguous free storage available.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2927I      GROUP *groupid* AUTHORIZED FOR:**

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the authorizations available for the members of the group.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2928I      USER *userid* AUTHORIZED FOR:**

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the authorizations available for the *userid*.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2929I      INTERACTION FOR IP SESSION**  
***ipcsid***

**Explanation**

The STATUS command outputs various information about the current processing environment. This message lists the IP sessions the user or group is authorized to interact with.

**System action**

The program continues.

**User response**

No action is required.

---

**GOM2930I      LAST TRIGGERED *date time***

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists a rule, a schedule, or a monitor's last trigger date and time.

**System action:**

Processing continues.

**User response**

No action is required.

---

**GOM2931I      LAST PROCESSING COMPLETE  
                 date time**

---

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists an action's last processing date and time.

**System action:**

Processing continues.

**User response**

No action is required.

---

**GOM2932I      MONITOR *resource-type resource-*  
                 id TO OCCUR *nn* TIME(S) WITHIN  
                 *nn* MINUTE(S)**

---

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the type, ID, and expected processing occurrences of a resource.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2933I      EVENT TYPE *types***

---

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the types of an event monitor.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2934I      EVENT CLASS *classes***

---

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the classes of an event monitor.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2935I      SUBTRACT *nnn* FROM LINE  
                 SUPPRESS COUNT**

---

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the line suppression value adjustment if the rule is triggered.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2936I      ADD *nnn* TO LINE SUPPRESS  
                 COUNT**

---

**Explanation:**

The STATUS command outputs various information about the current execution environment. This message lists the line suppression value adjustment if the rule is triggered.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2937I      SET LINE SUPPRESS COUNT TO  
                 *nnn***

---

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the line suppression value adjustment if the rule is triggered.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2938I      SUPPRESS NEXT *nnn* LINES**

---

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the current value of the line suppression count for a console.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2939I      ESMAPREF(*esmprefix*)**

---

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the current value of DEFOPTN ESMPREF.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2940I      PROCESSING ENV OVERRIDE *env***

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists if a rule, schedule, or monitor provides a processing environment server override.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2941I      PROCESSING ENV SERVER CAN  
BE OVERRIDDEN**

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists if an action processing environment can be overridden.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2942I      PROCESSING SUPPRESSED *count*  
TIME(S)**

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists how many times a rule, schedule, or monitor processing was suppressed due to LIMIT.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2943I      PROCESSING LIMIT IS *count*  
OCCURRENCE(S) WITHIN *interval*  
*nnn***

**Explanation**

The STATUS command outputs various information about the current processing environment. This message lists the *count* and *interval* processing limits of a rule, schedule, or monitor. The interval amount, *nnn*, can be any of the following: .

- *nnn* DAY(S), *nn* HOUR(S), *nn* MINUTE(S), *nn* SECOND(S)
- *nn* HOUR(S), *nn* MINUTE(S), *nn* SECOND(S)
- *nn* MINUTE(S), *nn* SECOND(S)
- *nn* SECOND(S)

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2945I      EXCLUDE GROUP *groupid***

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the excluded group for a rule or event monitor.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2946I      EXCLUDE USER *userid***

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the excluded user for a rule or event monitor.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2947I      AUTHCMD Y|N**

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the current setting for DEFOPTN AUTHCMD.

**System action:**

The program continues.

## User response

No action is required.

---

**GOM2948I**      **SYMBOLS** *abcd*

### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the current setting for DEFOPTN SYMBOLS.

### System action:

The program continues.

## User response

No action is required.

---

**GOM2949I**      **DSTADJ** Y|N

### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the current setting for DEFOPTN DSTADJ.

### System action:

The program continues.

## User response

No action is required.

---

**GOM2950I**      **MIXCASE** Y|N

### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the current setting for DEFOPTN MIXCASE.

### System action:

The program continues.

## User response

No action is required.

---

**GOM2951I**      **HLDATTR** *tttt,a,h,xat,clr*

### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the setting for DEFOPTN HLDATTR.

### System action:

The program continues.

## User response

No action is required.

---

**GOM2952I**      **ESM** Y|N

### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the current setting for DEFOPTN ESM.

### System action:

The program continues.

## User response

No action is required.

---

**GOM2953I**      **ALTERNATE SECLABEL** *seclabel*

### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the ALTSECLB for an action.

### System action:

The program continues.

## User response

No action is required.

---

**GOM2954I**      **ALTERNATE USERID** *userid*

### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the ALTUSER for an action.

### System action:

The program continues.

## User response

No action is required.

---

**GOM2955I**      **EXCLUDE "string" BETWEEN**  
**COLUMNS** *begcol* **AND** *endcol*

### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists string matching exclusion information for a rule.

### System action:

The program continues.

## User response

No action is required.

---

**GOM2956I**      **MATCH "string" BETWEEN**  
**COLUMNS** *begcol* **AND** *endcol*

### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists string matching information for a rule.



**System action:**  
The program continues.

### User response

No action is required.

---

**GOM2957I**      **GROUP** *groupid*

**Explanation:**  
The STATUS command outputs various information about the current processing environment. This message lists *groupid* information for a rule or event monitor.

**System action:**  
The program continues.

### User response

No action is required.

---

**GOM2958I**      **USER** *userid*

**Explanation:**  
The STATUS command outputs various information about the current processing environment. This message lists *userid* information for a rule or event monitor.

**System action:**  
The program continues.

### User response

No action is required.

---

**GOM2959I**      **INTERVAL WHEN** *interval*

**Explanation:**  
The STATUS command outputs various information about the current processing environment. This message appears when using the STATUS DETAIL SCHEDULE command and lists interval information from the input on the DEFSCHD command.

**System action:**  
The program continues.

### User response

No action is required.

---

**GOM2960I**      **INTERVAL EVERY** *interval*

**Explanation:**  
The STATUS command outputs various information about the current processing environment. This message appears when using the STATUS DETAIL SCHEDULE command and lists interval information from the input on the DEFSCHD command.

**System action:**

The program continues.

### User response

No action is required.

---

**GOM2961I**      **NEXT RUN** *CCYY/MM/DD hh:mm* |  
**NEXT POSSIBLE EXECUTION AT**  
*CCYY/MM/DD hh:mm*

### Explanation

The STATUS command displays information about the current processing environment. The time and date that each resource will next be evaluated for processing is shown. The evaluation process determines if any exclusionary condition exists for the resource. If no exclusionary condition exists, the action associated with the resource is scheduled for processing. Otherwise, the associated action is not scheduled for processing. In either case, the next evaluation date and time is incremented by the interval associated with the resource.

The exclusionary conditions which may restrict the resource and prevent it from being executed are:

- It has been suspended by the SUSPEND command.
- It has exceeded its LIMIT operand.
- For schedule resources only, other active schedules defined in the CONFLICT operand prevent its execution.
- For machine monitor resources only, other VM machine conditions defined in the DEPEND operand prevent its execution.
- The system name is/is not in the group defined by the SYSGRP/EXSYSGRP operand.
- The date and time fall within a holiday and is not allowed by the DAYS operand and DEFDATE command.
- The day of the week is not allowed by the DAYS operand.
- The hour of the day is not allowed by the HOURS operand.

**System action:**  
The program continues.

### User response

No action is required.

---

**GOM2962I**      **CURRENTLY ACTIVE**

**Explanation:**  
The STATUS command outputs various information about the current processing environment. This message tells if a schedule is active.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2963I      MAXIMUM DEFERRAL *nn***
**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the maximum deferral time information for a schedule.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2964I      *schdid***
**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists conflicting schedules for a schedule.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2965I      CONFLICTS WITH SCHEDULE(S) :**
**Explanation:**

The STATUS command outputs various information about the current processing environment. This message is followed by the conflicts for a schedule.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2966I      INCREASE LOW *low* HIGH *high***
**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the low and high increase values for a spool or page monitor.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2967I      USAGE LOW *low* HIGH *high***
**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the low and high usage values for a spool or page monitor.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2968I      NEXT CHECK *CCYY/MM/DD hh:mm*  
| NEXT POSSIBLE EXECUTION AT  
*CCYY/MM/DD hh:mm***
**Explanation**

The STATUS command displays information about the current processing environment. The time and date that each resource will next be evaluated for processing is shown. The evaluation process determines if any exclusionary condition exists for the resource. If no exclusionary condition exists, the action associated with the resource is scheduled for processing. Otherwise, the associated action is not scheduled for processing. In either case, the next evaluation date and time is incremented by the interval associated with the resource.

The exclusionary conditions which may restrict the resource and prevent it from being executed are:

- It has been suspended by the SUSPEND command.
- It has exceeded its LIMIT operand.
- For schedule resources only, other active schedules defined in the CONFLICT operand prevent its execution.
- For machine monitor resources only, other VM machine conditions defined in the DEPEND operand prevent its execution.
- The system name is/is not in the group defined by the SYSGRP/EXSYSGRP operand.
- The date and time fall within a holiday and is not allowed by the DAYS operand and DEFDATE command.
- The day of the week is not allowed by the DAYS operand.
- The hour of the day is not allowed by the HOURS operand.

**System action:**

The program continues.

## User response

No action is required.

---

<b>GOM2969I</b>	<b>DEPENDS ON USER <i>userid</i> STS <i>status (userid status) {connector}</i></b>
-----------------	--

### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the dependency information for a machine monitor.

### System action:

The program continues.

## User response

No action is required.

---

<b>GOM2970I</b>	<b>RETRY INTERVAL <i>nn</i> MINUTE(S) RETRY COUNT <i>nn</i></b>
-----------------	---

### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the retry information for a machine monitor.

### System action:

The program continues.

## User response

No action is required.

---

<b>GOM2971I</b>	<b>DELAY <i>nn</i> MINUTE(S)</b>
-----------------	----------------------------------

### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the delay information for a machine monitor.

### System action:

The program continues.

## User response

No action is required.

---

<b>GOM2972I</b>	<b>INTERVAL <i>nn</i> MINUTE(S)</b>
-----------------	-------------------------------------

### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the interval information for a resource.

### System action:

The program continues.

## User response

No action is required.

---

<b>GOM2973I</b>	<b>USER <i>userid (userid status)</i></b>
-----------------	---

### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the user information for a machine monitor.

### System action:

The program continues.

## User response

No action is required.

---

<b>GOM2974I</b>	<b>PARM "<i>parm</i>"</b>
-----------------	---------------------------

### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the parameter information for a resource.

### System action:

The program continues.

## User response

No action is required.

---

<b>GOM2975I</b>	<b>HOURS <i>hours eligible for processing</i></b>
-----------------	---

### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the hours information for a resource.

### System action:

The program continues.

## User response

No action is required.

---

<b>GOM2976I</b>	<b>DAYS <i>days eligible for processing</i></b>
-----------------	---

### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the days information for a resource.

### System action:

The program continues.

## User response

No action is required.

---

<b>GOM2977I</b>	<b>ACTION <i>actnid</i></b>
-----------------	-----------------------------

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the action ID information for a resource.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2978I      SUSPENDED****Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists if a resource is suspended from processing.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2979I      NEXT ACTION *actnid* DELAY *nn*  
MINUTES *nn* SECONDS****Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the next action information for an action.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2980I      COMMAND "*command*"****Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the command information for an action.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2981I      *attribute*****Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the input or output actions for an action.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2982I      OUTPUT ACTIONS:****Explanation:**

The STATUS command outputs various information about the current processing environment. This message is followed by the output actions for an action.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2983I      INPUT ACTIONS:****Explanation:**

The STATUS command outputs various information about the current processing environment. This message is followed by the input actions for an action.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2984I      PROCESSING ENV SERVER *env*****Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the processing environment server for an action command.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2985I      LAST RC=*rc* ELAPSED TIME=*time*  
*in ms*****Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the return code and elapsed time of the last action command.

**System action:**

The program continues.

## User response

No action is required.

---

**GOM2986I** *wrkgrpid*

### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the work groups eligible to execute an action command.

### System action:

The program continues.

## User response

No action is required.

---

**GOM2987I** **PARTICIPATING IN PROCESSING ENV WORK GROUP(S):**

### Explanation:

The STATUS command outputs various information about the current processing environment. This message is followed by the work groups eligible to process an action command.

### System action:

The program continues.

## User response

No action is required.

---

**GOM2988I** **PROCESSING ENV UNIQUE**

### Explanation:

The STATUS command outputs various information about the current processing environment. This message is listed if a service machine may only process actions with an *ENV* matching the service name.

### System action:

The program continues.

## User response

No action is required.

---

**GOM2989I** **PROCESSING ACTION *actnid***

### Explanation:

The STATUS command outputs various information about the current processing environment. This message is listed if a service is currently processing an action.

### System action:

The program continues.

## User response

No action is required.

---

**GOM2990I** **NOT READY**

### Explanation:

The STATUS command outputs various information about the current processing environment. This message is listed if a service is not ready to process actions.

### System action:

The program continues.

### User response:

Determine why the service machine is not logged on or not executing GOMSVM.

---

**GOM2991I** **READY**

### Explanation:

The STATUS command outputs various information about the current processing environment. This message is listed if a service is ready to process actions.

### System action:

The program continues.

## User response

No action is required.

---

**GOM2992I** **SPOOL FOR *pattern***

### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the spool files a user or group is authorized to view.

### System action:

The program continues.

## User response

No action is required.

---

**GOM2993I** **CONSOLE FOR *pattern***

### Explanation:

The STATUS command outputs various information about the current processing environment. This message lists the consoles a user or group is authorized to view.

### System action:

The program continues.

## User response

No action is required.

---

**GOM2994I** **CONFIGURATION COMMANDS**

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists if a user is authorized for configuration commands.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2995I            CONTROL COMMANDS**


---

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists if a user or group is authorized for control commands.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2996I            EXTERNAL SECURITY MANAGER  
                         IN USE**


---

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists if an External Security Manager is in use for authorizations.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2997I            *userid***


---

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message lists the users viewing a console.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2998I            BEING VIEWED BY:**


---

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message is followed by a list of users viewing a console.

**System action:**

The program continues.

**User response**

No action is required.

---

**GOM2999I            STATUS DETAIL COMPLETE**


---

**Explanation:**

The STATUS command outputs various information about the current processing environment. This message is listed when the STATUS DETAIL command output is complete.

**System action:**

The program continues.

**User response**

No action is required.

## 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:

*Intellectual Property Licensing  
Legal and Intellectual Property Law  
IBM Japan Ltd.  
19-21, Nihonbashi-Hakozakicho, Chuo-ku  
Tokyo 103-8510, Japan*

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

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

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

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

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

*IBM 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 measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

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

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

#### COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

## Trademarks

---

IBM, the IBM logo, and [ibm.com](http://www.ibm.com)® are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at <http://www.ibm.com/legal/copytrade.shtml>.

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

## Terms and conditions for product documentation

---

Permissions for the use of these publications are granted subject to the following terms and conditions:

**Applicability:** These terms and conditions are in addition to any terms of use for the IBM website.

**Personal use:** You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

**Commercial use:** You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.



**Rights:** Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

## Privacy policy considerations

---

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user, or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth below.

This Software Offering does not use cookies or other technologies to collect personally identifiable information.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM's Privacy Policy at <http://www.ibm.com/privacy> and IBM's Online Privacy Statement at <http://www.ibm.com/privacy/details> the section entitled "Cookies, Web Beacons and Other Technologies" and the "IBM Software Products and Software-as-a-Service Privacy Statement" at <http://www.ibm.com/software/info/product-privacy>.



---

# Index

## A

- accessing
  - consoles [23](#)
  - spools [23](#)
- accounting
  - files [193](#)
  - records [193](#)
- Action commands
  - substitution variables [183](#)
- action processing
  - defining servers [24](#)
- actions
  - defining [84](#)
  - deleting [66](#), [124](#)
  - processing [38](#)
- adding comments [22](#), [151](#)
- ADDR operand
  - using DSN [197](#)
- ALTRCON command [148](#)
- ALTRCRON
  - Operands [148](#)
- assigning
  - authority [22](#), [43](#)
- AUTH
  - external security [24](#)
  - Operands [43](#)
- AUTH command [22](#), [43](#)
- authority
  - assign [43](#)
  - assigning [22](#)
  - IP session [24](#)
  - revoking [74](#)

## B

- basics of product [1](#)

## C

- captured console data [50](#)
- CMD
  - Operands [46](#)
- CMDIPCS
  - Operands [151](#)
- CMDIPCS command [150](#)
- CMS command [45](#)
- collecting
  - remote file input [202](#)
  - syslog data [200](#)
- command shortcuts [28](#)
- commands
  - ALTRCON [148](#)
  - configuration
    - CONFIG [46](#), [81](#)
    - DEFACTN [84](#)
    - DEFDATE [89](#)

- commands (*continued*)

- configuration (*continued*)

- DEFEMON [91](#)
    - DEFGROUP [99](#)
    - DEFIMON [95](#)
    - DEFMMON [101](#)
    - DEFPMON [105](#)
    - DEFRULE [109](#)
    - DEFSCHD [115](#)
    - DEFMON [120](#)
    - DEFACTN [124](#)
    - DEFACTQ [66](#)
    - DELDATE [125](#)
    - DELEMON [126](#)
    - DELGROUP [127](#)
    - DELMON [129](#)
    - DELPMON [130](#)
    - DELRULE [131](#)
    - DELSCHD [132](#)
    - DELSMON [133](#)
    - RESUME [134](#)
    - RUN [137](#)
    - STATUS [76](#), [139](#)
    - SUSPEND [143](#)
  - console access [147](#)
  - control
    - AUTH [43](#)
    - CMS [45](#)
    - CONFIG [46](#), [81](#)
    - CP [48](#)
    - DEFDSPS [50](#)
    - DEFOPTN [57](#)
    - DEFSRVM [63](#)
    - DEFVIEW [64](#)
    - DELDSPS [68](#)
    - DELSRVM [70](#)
    - DELVIEW [71](#)
    - QUIT [73](#)
    - REVOKE [74](#)
    - STATUS [76](#), [139](#)
    - STOP [80](#)
    - VIEWLOG [159](#)
  - DEFIPCS [200](#)
  - entering [31](#)
  - GOMRSIF [203](#)
  - issue dynamically [2](#)
  - issuing dynamically [36](#)
  - log access [147](#)
  - LOGTEXT [151](#)
  - rules [31](#)
  - spool access [147](#)
  - substitution variables [183](#)
  - syntax [31](#)
  - system management
    - CMDIPCS [150](#)
    - DEFIPCS [52](#)
    - DELIPCS [69](#)

- commands (*continued*)
  - system management (*continued*)
    - GOMCMD [36](#)
    - GOMMAIN [35](#)
    - GOMSVM [38](#)
    - GOMTDM [39](#)
  - VIEWCON [153](#)
  - IEWSPL [165](#)
- comments
  - adding [151](#)
- components
  - message processing [2](#)
- CONFIG command [46](#), [81](#)
- configuration
  - authority [23](#)
- configuration commands [81](#)
- configuration files
  - managing [28](#)
- configuring
  - Operations Manager [5](#)
- console commands [147](#)
- console data
  - verifying [10](#)
- console data space names [147](#)
- consoles
  - accessing [23](#)
  - viewing [153](#)
- control
  - authority [23](#)
- control commands [43](#)
- controlling
  - Operations Manager [36](#)
- copying
  - sample configuration file [21](#)
- CP command [48](#)
- CP directory [6](#)
- customizing
  - log file management [195](#)

## D

- data space names [147](#)
- DEFACTN
  - Operands [85](#)
- DEFACTN command [84](#)
- DEFDATE
  - Operands [89](#)
- DEFDATE command [89](#)
- DEFDSPS
  - Operands [50](#)
- DEFDSPS command [50](#)
- DEFEMON
  - substitution variables [183](#)
- DEFEMON command [91](#)
- DEFGROUP command [99](#)
- DEFIMON
  - Operands [95](#)
  - substitution variables [183](#)
- DEFIMON command [95](#)
- defining
  - action processing servers [24](#)
  - actions [84](#)
  - configuration authority [23](#)
  - console access [23](#)

- defining (*continued*)
  - control authority [23](#)
  - event monitors [91](#)
  - groups [99](#)
  - holidays [89](#)
  - monitored virtual machines [101](#)
  - multiple views [64](#)
  - page space monitors [105](#)
  - rules [109](#)
  - schedules [115](#)
  - security options [57](#)
  - spool access [23](#)
  - spool usage monitors [120](#)
  - storage for captured data [50](#)
  - system options [24](#)
  - views [64](#)
- DEFIPCS
  - DSN names [197](#)
  - Operands [52](#)
- DEFIPCS command [52](#), [200](#)
- DEFMMON
  - substitution variables [183](#)
- DEFMMON command [101](#)
- DEFMMON example [104](#)
- DEFOPTN
  - ESM(Y/N) [24](#)
  - operands [57](#)
- DEFOPTN command [57](#)
- DEFPMON
  - Operands [106](#)
  - substitution variables [183](#)
- DEFPMON command [105](#)
- DEFRULE
  - substitution variables [183](#)
- DEFRULE command [109](#)
- DEFSCHD
  - Operands [116](#)
  - substitution variables [183](#)
- DEFSCHD command [115](#)
- DEFSMON
  - Operands [120](#)
- DEFSMON command [120](#)
- DEFSRVM command [24](#), [63](#)
- DEFVIEW command [64](#)
- DEFACTN command [124](#)
- DEFACTQ
  - Operands [66](#)
- DEFACTQ command [66](#)
- DELDATE command [125](#)
- DELDSPS command [68](#)
- DELEMON
  - Operands [126](#)
- DELEMON command [126](#)
- deleting
  - actions [66](#), [124](#)
  - authority [74](#)
  - event monitors [126](#)
  - groups [127](#)
  - holidays [125](#)
  - monitored virtual machine [129](#)
  - page space monitors [130](#)
  - rules [131](#)
  - schedules [132](#)
  - service machines [70](#)

- deleting (*continued*)
  - spool usage monitors
    - deleting [133](#)
    - storage definition [68](#)
    - views [71](#)
- DELGROUP command [127](#)
- DELIMON
  - Operands [128](#)
- DELIPCS
  - Operands [69](#)
- DELIPCS command [69](#)
- DELIPCS example [70](#)
- DELMMON
  - Operands [129](#)
- DELMMON command [129](#)
- DELMMON example [130](#)
- DELPMON
  - Operands [130](#)
- DELPMON command [130](#)
- DELPMON example [131](#)
- DELRULE
  - Operands [131](#)
- DELRULE command [131](#)
- DELSCHD command [132](#)
- DELSMON command [133](#)
- DELSRVM command [70](#)
- DELSRVM example [71](#)
- DELVIEW command [71](#)
- directories
  - CP [9](#)
- DSN name resolution [197](#)

## E

- entering
  - commands [31](#)
- error messages [215](#)
- ESM class
  - profiles [11](#)
- ESMCLASS
  - setting [11](#)
- ESMPREF
  - setting [11](#)
- event monitors
  - defining [91](#)
  - deleting [126](#)
- examples
  - adding comments [22](#)
  - assigning authority [22](#)
  - AUTH command [22](#)
  - defining action processing servers [24](#)
  - DEFSRVM [24](#)
  - GOMCMD command [37](#)
  - STATUS command [77](#), [140](#)
- Examples
  - DEFMMON [104](#)
  - DELIPCS [70](#)
  - DELMMON [130](#)
  - DELPMON [131](#)
  - DELSRVM [71](#)
- external security
  - AUTH [24](#)
- external security manager
  - class [11](#)

- external security manager (*continued*)
  - profile prefix [11](#)
- External Security Manager (ESM) [10](#), [57](#)

## F

- files
  - accounting [193](#)
  - configuration [5](#)
  - MAS COMMANDS [35](#)
  - monitoring [194](#)
  - OPMGRM1 CONFSAMP [2](#), [21](#), [35](#)
  - PROFILE EXEC [9](#), [27](#)
  - PROFM1 SAMPEXEC [5](#)
  - PROFS1 SAMPEXEC [7](#)
  - PROFUSR SAMPEXEC [8](#)
  - sending multiple [202](#)
  - startup configuration [21](#), [27](#)

## G

- GOMCMD [2](#)
- GOMCMD command [36](#)
- GOMGLBL
  - preparing [209](#)
- GOMGLBL facility [209](#)
- GOMMAIN
  - stopping [73](#)
- GOMMAIN command [35](#), [36](#)
- GOMRSIF command [203](#)
- GOMSVM [2](#)
- GOMSVM command [38](#)
- GOMTDM command [39](#)
- groups
  - defining [99](#)
  - deleting [127](#)

## H

- hardware
  - requirements [3](#)
- Help menu [3](#)
- holidays
  - defining [89](#)
  - deleting [125](#)

## I

- idle monitors
  - defining [95](#)
- IDs
  - administrator [8](#)
  - monitored by Operations Manager [9](#)
  - running GOMMAIN [5](#), [9](#)
  - running GOMSVM [7](#)
  - using GOMCMD [8](#)
  - using GOMTDM [8](#)
- installing
  - Operations Manager [5](#)
- IP
  - apps authority [24](#)
- IPV4 session [197](#)
- IPV6 session [197](#)

## L

- legal notices
  - notices [263](#)
  - trademarks [264](#)
- links
  - non-IBM Web sites [264](#)
- log file
  - adding comments [151](#)
  - management [195](#)
- LOGTEXT command [151](#)

## M

- managing
  - configuration files [28](#)
  - log files [195](#)
- MAS COMMANDS file [35](#)
- matching wildcard patterns [181](#)
- messages
  - processing [35](#)
- modifying
  - startup configuration file [27](#)
- monitoring
  - files [194](#)
  - Operations Manager [36](#)
  - records [194](#)
  - virtual machines [101](#)

## N

- notices [263](#)

## O

- Operations Manager
  - automating sessions [39](#), [52](#), [150](#)
  - benefits [1](#)
  - component interaction [2](#)
  - components [2](#)
  - deleting sessions [69](#)
  - error messages [215](#)
  - log management [195](#)
  - monitoring [36](#)
  - processing
    - actions [38](#)
    - status [76](#), [139](#)
    - user interaction [2](#)
    - what is [1](#)
- OPMGRM1 [6](#)
- OPMGRM1 CONFSAMP file [21](#), [35](#)

## P

- page space monitors
  - defining [105](#)
  - deleting [130](#)
- pattern matching for wildcards [181](#)
- preparing
  - GOMGLBL [209](#)
- processing
  - messages [35](#)

- processing (*continued*)
  - remote file input [202](#)
  - syslog data [200](#)
- PROFILE EXEC [7](#)
- PROFILE EXEC file [9](#)
- profiles
  - ESM class [11](#)
- PROFM1 SAMPEXEC file [5](#)
- PROFS1 SAMPEXEC file [7](#)
- PROFUSR SAMPEXEC file [8](#)

## Q

- QUIT command [73](#)

## R

- RACF [10](#)
- records
  - accounting [193](#)
  - monitoring [194](#)
- remote file input
  - collecting [202](#)
  - processing [202](#)
- requirements
  - hardware [3](#)
  - software [3](#)
- resolving domain names [197](#)
- RESUME command [134](#)
- REVOKE command [74](#)
- revoking
  - authority [74](#)
- REXX EXECs [28](#)
- rules
  - defining [109](#)
  - deleting [131](#)
- RUN
  - substitution variables [183](#)
- RUN command [137](#)

## S

- sample configuration file
  - copying [21](#)
- sample REXX EXECs [28](#)
- schedules
  - defining [115](#)
  - deleting [132](#)
- security management [57](#)
- sending
  - file content [202](#)
  - multiple files [202](#)
- servers
  - action processing [7](#), [24](#)
- service machines
  - deleting [70](#)
  - starting [27](#)
- sessions
  - automating [39](#), [52](#), [150](#)
  - deleting [69](#)
- setting
  - ESMCLASS [11](#)
  - ESMPREF [11](#)

- shortcuts to common commands [28](#)
- software
  - requirements [3](#)
- spool access commands [147](#)
- spool files
  - viewing [165](#)
- spool usage monitors
  - defining [120](#)
- spools
  - accessing [23](#)
- starting
  - service machines [27](#)
- STATUS command [76](#), [139](#)
- STOP command [80](#)
- stopping
  - GOMMAIN [73](#)
- storage definition
  - creating [50](#)
  - deleting [68](#)
- substitution variables [183](#)
- SUSPEND command [143](#)
- syntax [31](#)
- syntax diagrams
  - how to read [213](#)
- syslog data
  - collecting [200](#)
  - processing [200](#)
  - translating [201](#)
- system management
  - commands [35](#)
- system options
  - defining [24](#)

## T

- tables
  - translation [201](#)
- TCP protocol [197](#)
- testing
  - configuration [205](#)
  - installation [205](#)
- trademarks [264](#)
- translating
  - syslog data [201](#)

## U

- UDP protocol [197](#)

## V

- variables
  - substitution [183](#)
- verifying
  - console data [10](#)
- VIEWCON
  - Operands [153](#)
- VIEWCON command [153](#)
- viewing
  - consoles [153](#)
  - file content [202](#)
  - log [159](#)
  - multiple consoles [64](#)

- viewing (*continued*)
  - multiple files [202](#)
  - spool files [165](#)
- viewing spool files [206](#)
- VIEWLOG
  - Operands [160](#)
- VIEWLOG command [159](#)
- views
  - defining multiple [64](#)
  - deleting [71](#)
- VIEWSPL command [165](#)
- virtual machine
  - remove monitoring [129](#)
- virtual machines
  - defining [101](#)
  - monitoring [101](#)

## W

- wildcard pattern matching [181](#)









Product Number: 5697-J10

SC18-9347-29

